

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problems Mailbox.**

PTO 2003-3009

S.T.I.C. Translations Branch

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-250054

(43) 公開日 平成11年(1999) 9月17日

(51) Int.Cl. ⁸	識別記号	F I	
G 0 6 F 17/24		G 0 6 F 15/20	5 3 4 Z
17/21			5 7 0 D
17/30			5 9 0 E
// G 0 6 T 11/60		15/40	3 7 0 A
		15/419	3 2 0
審査請求 未請求 請求項の数20 O L (全 18 頁) 最終頁に続く			

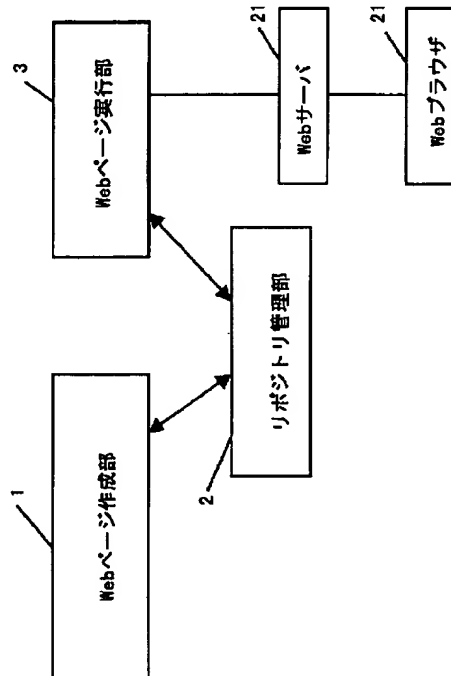
(21) 出願番号	特願平10-49508	(71) 出願人	000006013 三菱電機株式会社 東京都千代田区丸の内二丁目2番3号
(22) 出願日	平成10年(1998) 3月2日	(72) 発明者	小坂 哲也 東京都千代田区丸の内二丁目2番3号 三 菱電機株式会社内
		(72) 発明者	藤本 真弓 東京都千代田区丸の内二丁目2番3号 三 菱電機株式会社内
		(72) 発明者	渡部 修介 東京都千代田区丸の内二丁目2番3号 三 菱電機株式会社内
		(74) 代理人	弁理士 宮田 金雄 (外2名)

(54) 【発明の名称】 切り張り型Webページ生成装置および生成方法

(57) 【要約】

【課題】 既存のWebページ上のオブジェクトを抽出し、そのオブジェクトを基に新たなWebページを自由なレイアウトで作成することによって、既存のWebページ上のオブジェクトの再利用を可能にする。

【解決手段】 複数のWebページからオブジェクトを抽出し、自由なレイアウトで新たなWebページを作成するWebページ作成部と、作成されたWebページの構成情報を格納すると共に、貼り付けられたオブジェクトを更新するリポジトリ管理部と、リポジトリ管理部に格納された構成情報とオブジェクトから実際にWebページを作成するWebページ実行部とを備え、抽出したオブジェクトおよび作成したWebページの構成をリポジトリ管理部に格納し、それを利用して、ユーザの要求時に動的にWebページを作成する。これによって、作成した切り張り型Webページ上では常に最新のオブジェクトが参照できる。



【特許請求の範囲】

【請求項1】 既存の複数のWebページを利用して新たなWebページを作成し参照するWebページ生成装置において、
複数のWebページからオブジェクトを抽出し、自由なレイアウトで新たなWebページを作成するWebページ作成部と、
作成されたWebページの構成情報を格納すると共に、貼り付けられたオブジェクトを更新するリポジトリ管理部と、
前記リポジトリ管理部に格納された構成情報とオブジェクトから実際にWebページを作成するWebページ実行部と、を備えることを特徴とするWebページ生成装置。

【請求項2】 請求項1記載のWebページ生成装置において：前記Webページ作成部は、
既存の複数のWebページからオブジェクトを抽出するオブジェクト抽出エディタと、
抽出されたオブジェクトをエディットするWebページ作成エディタとを備えたことを特徴とするWebページ生成装置。

【請求項3】 請求項2記載のWebページ生成装置において：前記オブジェクト抽出エディタは、
入力されたWebページをオブジェクト毎に分割する機能を持つHTML解析部と、
前記HTML解析部で分割されたオブジェクトの分割の仕方を修正する機能を持つオブジェクト修正部とを備え、
既存のWebページからオブジェクトを抽出することを特徴とするWebページ生成装置。

【請求項4】 請求項3記載のWebページ生成装置において：前記HTML解析部は、
オブジェクトをWebページから抽出するためのルールを格納する解析ルールDBと、
前記解析ルールDBに格納されたルールに従って、入力されたWebページを与えられたオブジェクトに分割するHTML解析エンジンと、
オブジェクトをWebページから抽出するためのルールを格納した解析ルールDBとを備えたことを特徴とするWebページ生成装置。

【請求項5】 請求項3記載のWebページ生成装置において：前記オブジェクト修正部は、
前記HTML解析部によってオブジェクトに分割されたWebページをユーザに表示しオブジェクトの分割の修正を行うエディタ機能およびオブジェクトの持つ属性情報を付加する機能を有するオブジェクトパッケージ部を備えたことを特徴とするWebページ生成装置。

【請求項6】 請求項1記載のWebページ生成装置において：前記Webページ作成エディタは、
リポジトリI/O部を介してリポジトリ管理部との間で

オブジェクトの情報や作成した切り張りWebページの構成情報の登録および取り出しを行い、抽出されたオブジェクトをグラフィカルに表示してユーザにWebページ作成のための部品として抽出されたオブジェクトを提供するオブジェクト表示部と、

前記オブジェクト表示部に表示されたオブジェクトを配置して切り張りWebページを作成するWebページレイアウト部とを備えたことを特徴とするWebページ生成装置。

10 【請求項7】 請求項1記載のWebページ生成装置において：前記リポジトリ管理部は、
抽出したWebページ上のオブジェクトおよびオブジェクトの属性情報を管理するためのオブジェクトDBと、
作成したWebページのレイアウト情報を管理する切り張りWebページ構成情報DBと、
オブジェクトの更新を行うオブジェクトチェックエンジンとを備えたことを特徴とするWebページ生成装置。

【請求項8】 請求項7記載のWebページ生成装置において：前記オブジェクトチェックエンジンは、
20 リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報の登録および取り出しを行い、取り出したオブジェクト情報の中の抽出ルールに従って、オブジェクトを解析し抽出しオブジェクト比較部に渡すHTML解析エンジンと、
リポジトリI/O部から渡されるリポジトリ管理部に登録されているオブジェクトおよびHTML解析エンジンから渡されるオリジナルのオブジェクトを比較し、オブジェクトが更新されていればリポジトリI/O部にリポジトリ管理部との間でオブジェクトの再登録を依頼するオブジェクト比較部とを備えたことを特徴とするWebページ生成装置。

30 【請求項9】 請求項1記載のWebページ生成装置において：前記Webページ実行部は、
リポジトリ管理部からURL名に対応したWebページの構成情報を取り出す実行制御部と、
前記実行制御部からの切り張りWebページ構成情報に基づいてWebページの作成を行うWebページリレイアウト部と、を備えたことを特徴とするWebページ生成装置。

40 【請求項10】 請求項9記載のWebページ生成装置において：前記Webページリレイアウト部は、
リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報および切り張りWebページの構成情報の取り出しを行い、取り出されたオブジェクトの情報、切り張りWebページのレイアウト情報をもとに、HTMLファイルを作成するHTML生成エンジンを備えたことを特徴とするWebページ生成装置。

【請求項11】 請求項9記載のWebページ生成装置において：前記実行制御部は、
50 リポジトリI/O部を介してリポジトリ管理部との間で

オブジェクトの情報および切り張りWebページの構成情報の取り出しを行い、取り出されたオブジェクトの情報および切り張りWebページの構成情報をもとに、リポジトリ管理部およびWebページレイアウト部の制御を行う実行マネージャとを備え、Webページの生成を制御することを特徴とするWebページ生成装置。

【請求項12】 既存の複数のWebページを利用して新たなWebページを作成し参照するWebページ生成方法において、

Webページ作成部は、複数のWebページからオブジェクトを抽出し、自由なレイアウトで新たなWebページを作成し、

リポジトリ管理部は、作成されたWebページの構成情報を格納すると共に、貼り付けられたオブジェクトを更新し、

Webページ実行部は、前記リポジトリ管理部に格納された構成情報とオブジェクトから実際にWebページを作成することを特徴とするWebページ生成方法。

【請求項13】 請求項12記載のWebページ生成方法において：前記Webページ作成部では、

オブジェクト抽出エディタが、既存の複数のWebページからオブジェクトを抽出し、

Webページ作成エディタが、抽出されたオブジェクトをエディットすることを特徴とするWebページ生成方法。

【請求項14】 請求項13記載のWebページ生成方法において：前記オブジェクト抽出エディタでは、HTML解析部が、入力されたWebページをオブジェクト毎に分割し、

オブジェクト修正部が、前記HTML解析部で分割されたオブジェクトの分割の仕方を修正し、既存のWebページからオブジェクトを抽出することを特徴とするWebページ生成方法。

【請求項15】 請求項14記載のWebページ生成方法において：前記HTML解析部では、解析ルールDBが、オブジェクトをWebページから抽出するためのルールを格納し、

HTML解析エンジンが、前記解析ルールDBに格納されたルールに従って、入力されたWebページを与えられたオブジェクトに分割することを特徴とするWebページ生成方法。

【請求項16】 請求項14記載のWebページ生成方法において：前記オブジェクト修正部では、オブジェクトパッケージ部が、前記HTML解析部によってオブジェクトに分割されたWebページをユーザに表示しオブジェクトの分割の修正を行い、およびオブジェクトの持つ属性情報を付加することを特徴とするWebページ生成方法。

【請求項17】 請求項13記載のWebページ生成方法において：前記Webページ作成エディタでは、

オブジェクト表示部が、リポジトリI/O部を介して、リポジトリ管理部との間でオブジェクトの情報や作成した切り張りWebページの構成情報の登録、取り出し等を行い、抽出されたオブジェクトをグラフィカルに表示してユーザにWebページ作成のための部品として抽出されたオブジェクトを提供し、Webページレイアウト部が、オブジェクト表示部に表示されたオブジェクトを配置して切り張りWebページを作成することを特徴とするWebページ生成方法。

【請求項18】 請求項12記載のWebページ生成方法において：前記リポジトリ管理部は、オブジェクトDBにおいて、抽出したWebページ上のオブジェクトおよび、オブジェクトの属性情報を管理し、

切り張りWebページ構成情報DBにおいて、作成したWebページのレイアウト情報を管理し、

オブジェクトチェックエンジンにおいて、オブジェクトの更新などを行うことを特徴とするWebページ生成方法。

20 【請求項19】 請求項18記載のWebページ生成方法において：前記オブジェクトチェックエンジンは、

HTML解析エンジンが、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報の登録、取り出しを行い、取り出したオブジェクト情報の中の抽出ルールに従って、オブジェクトを解析し抽出し、オブジェクト比較部が、リポジトリI/O部から渡されるリポジトリ管理部に登録されているオブジェクトおよびHTML解析エンジンから渡されるオリジナルのオブジェクトを比較し、オブジェクトが更新されていればリポジトリI/O部を介してリポジトリ管理部にオブジェクトの再登録を依頼することを特徴とするWebページ生成方法。

【請求項20】 請求項12記載のWebページ生成方法において：前記Webページ実行部は、

実行マネージャが、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報、切り張りWebページの構成情報の取り出しを行い、取り出されたオブジェクトの情報、切り張りWebページの構成情報をもとに、リポジトリ管理部、HTML生成エンジンの制御を行い、

前記HTML生成エンジンが、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報および切り張りWebページの構成情報の取り出しを行い、取り出された情報をもとに、HTMLファイルを作成することを特徴とするWebページ生成方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、インターネット、イントラネット上に提供されるWebページ中のオブジェクト（図、表、アプレット、文字列等）を再利用し、

新たなWebページを生成し、実行するための装置および方法に関するものである。

【0002】

【従来の技術】インターネット、イントラネットに散在する独立した情報がある目的の元に関連付けを行う場合、従来は図21のような「リンク集」と呼ばれるWebページを作成する方法がある。この方法によれば、散在するWebページ101、102、103のURL情報をそれぞれ、ハイパーリンクとして記述したリンク集Webページ104を作成して、相互をリンクする。ユーザはこのリンク集Webページ104上のハイパーリンクのタグを選択することにより、散在するWebページ101、102、103をWebブラウザ上に表示する。

【0003】

【発明が解決しようとする課題】通常、ユーザが求める情報の単位はそれぞれのWebページそのものということではなく、Webページ上のあるオブジェクトである。しかしながら、従来の「リンク集」による情報の関連付けは、ユーザが必要とする情報(オブジェクト)を自由なレイアウトで見ることが出来ないばかりか、必要とする情報を見るためには関連付けられた全てのWebページをブラウザ上で開いてみなければならない、という問題点があった。

【0004】本発明は上記のような問題点を解決するためになされたもので、インターネット、イントラネットに散在しユーザが必要とする情報(オブジェクト)をユーザの好みのレイアウトで参照可能にすることを目的とする。

【0005】

【課題を解決するための手段】第1の発明によれば、本発明は、複数のWebページからオブジェクトを抽出し、自由なレイアウトで新たなWebページを作成するWebページ作成部と、作成されたWebページの構成情報を格納すると共に、貼り付けられたオブジェクトを更新するリポジトリ管理部と、前記リポジトリ管理部に格納された構成情報とオブジェクトから実際にWebページを作成するWebページ実行部とから構成される。

【0006】第2の発明によれば、本発明のWebページ作成部は、既存の複数のWebページからオブジェクトを抽出するオブジェクト抽出エディタと、抽出されたオブジェクトをエディットするWebページ作成エディタとを備えるように構成される。

【0007】第3の発明によれば、本発明のオブジェクト抽出エディタは、入力されたWebページをオブジェクト毎に分割する機能を持つHTML解析部と、HTML解析部で分割されたオブジェクトの分割の仕方を修正する機能を持つオブジェクト修正部とを備え、既存のWebページからオブジェクトを抽出するように構成される。

【0008】第4の発明によれば、本発明のHTML解析部は、入力されたWebページを与えられたオブジェクトの分割のルールに従って、オブジェクトに分割することをHTML解析エンジンと、オブジェクトをWebページから抽出するためのルールを格納した解析ルールDBとを備えるように構成される。

【0009】第5の発明によれば、本発明のオブジェクト修正部は、HTML解析部によってオブジェクトに分割されたWebページをユーザに表示しオブジェクトの分割の修正を可能にするエディタ機能およびオブジェクトの持つ属性情報を付加する機能を有するオブジェクトパッケージ部を備える構成される。

【0010】第6の発明によれば、本発明のWebページ作成エディタは、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報の作成した切り張りWebページの構成情報の登録および取り出しを行い、抽出されたオブジェクトをグラフィカルに表示してユーザにWebページ作成のための部品として抽出されたオブジェクトを提供するオブジェクト表示部と、オブジェクト表示部に表示されたオブジェクトを配置して切り張りWebページを作成するWebページレイアウト部とを備えるように構成される。

【0011】第7の発明によれば、本発明のリポジトリ管理部は、抽出したWebページ上のオブジェクトおよびオブジェクトの属性情報を管理するためのオブジェクトDBと、作成したWebページのレイアウト情報を管理する切り張りWebページ構成情報DBと、オブジェクトの更新などを行うオブジェクトチェックエンジンとを備えるように構成される。

【0012】第8の発明によれば、本発明のオブジェクトチェックエンジンは、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報の登録および取り出しを行い、取り出したオブジェクト情報の中の抽出ルールに従って、オブジェクトを解析し抽出しオブジェクト比較部に渡すHTML解析エンジンと、リポジトリI/O部から渡されるリポジトリ管理部に登録されているオブジェクトおよびHTML解析エンジンから渡されるオリジナルのオブジェクトを比較し、オブジェクトが更新されていればリポジトリI/O部にリポジトリ管理部との間でオブジェクトの再登録を依頼するオブジェクト比較部とを備えるように構成される。

【0013】第9の発明によれば、本発明のWebページ実行部は、リポジトリ管理部からURL名に対応したWebページの構成情報を取り出す実行制御部と、前記実行制御部からの切り張りWebページ構成情報に基づいてWebページの作成を行うWebページリレイアウト部とを備えるように構成される。

【0014】第10の発明によれば、本発明のWebページ実行部は、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報の登録および切り張りWeb

bページの構成情報の取り出しを行い、取り出されたオブジェクトの情報および切り張りWebページのレイアウト情報をもとに、HTMLファイルを作成するHTML生成エンジンを備えるように構成される。

【0015】第11の発明によれば、本発明の実行制御部は、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報および切り張りWebページの構成情報の取り出しを行い、取り出されたオブジェクトの情報および切り張りWebページの構成情報をもとに、リポジトリ管理部およびWebページレイアウト部の制御を行う実行マネージャとを備え、Webページの生成を制御するように構成される。

【0016】第12の発明によれば、本発明は、Webページ作成部が、複数のWebページからオブジェクトを抽出し、自由なレイアウトで新たなWebページを作成し、リポジトリ管理部が、作成されたWebページの構成情報を格納すると共に、貼り付けられたオブジェクトを更新し、Webページ実行部が、前記リポジトリ管理部に格納された構成情報とオブジェクトから実際にWebページを作成するように構成される。

【0017】第13の発明によれば、本発明は、Webページ作成部では、オブジェクト抽出エディタが、既存の複数のWebページからオブジェクトを抽出し、Webページ作成エディタが、抽出されたオブジェクトをエディットするように構成される。

【0018】第14の発明によれば、本発明は、オブジェクト抽出エディタでは、HTML解析部が、入力されたWebページをオブジェクト毎に分割し、オブジェクト修正部が、前記HTML解析部で分割されたオブジェクトの分割の仕方を修正することによって、既存のWebページからオブジェクトを抽出するように構成される。

【0019】第15の発明によれば、本発明は、HTML解析部では、解析ルールDBが、オブジェクトをWebページから抽出するためのルールを格納し、HTML解析エンジンが、前記解析ルールDBに格納されたルールに従って、入力されたWebページを与えられたオブジェクトに分割するように構成される。

【0020】第16の発明によれば、本発明は、前記オブジェクト修正部では、オブジェクトパッケージ部が、前記HTML解析部によってオブジェクトに分割されたWebページをユーザに表示しオブジェクトの分割の修正を行い、およびオブジェクトの持つ属性情報を付加するように構成される。

【0021】第17の発明によれば、本発明は、Webページ作成エディタでは、オブジェクト表示部がリポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報や作成した切り張りWebページの構成情報の登録および取り出し等を行い、抽出されたオブジェクトをグラフィカルに表示してユーザにWebページ作

成のための部品として抽出されたオブジェクトを提供し、Webページレイアウト部がオブジェクト表示部に表示されたオブジェクトを配置して切り張りWebページを作成するように構成される。

【0022】第18の発明によれば、本発明は、リポジトリ管理部では、オブジェクトDBが、抽出したWebページ上のオブジェクトおよび、オブジェクトの属性情報を管理し、切り張りWebページ構成情報DBが、作成したWebページのレイアウト情報を管理し、オブジェクトチェックエンジンが、オブジェクトの更新を行うように構成される。

【0023】第19の発明によれば、本発明は、オブジェクトチェックエンジンでは、HTML解析エンジンが、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報の登録、取り出しを行い、取り出したオブジェクト情報の中の抽出ルールに従って、オブジェクトを解析し抽出し、オブジェクト比較部が、リポジトリI/O部から渡されるリポジトリ管理部に登録されているオブジェクトおよびHTML解析エンジンから渡されるオリジナルのオブジェクトを比較し、オブジェクトが更新されていればリポジトリI/O部を介してリポジトリ管理部にオブジェクトの再登録を依頼するように構成される。

【0024】第20の発明によれば、本発明は、Webページ実行部では、実行マネージャが、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報、切り張りWebページの構成情報の取り出しを行い、取り出されたオブジェクトの情報、切り張りWebページの構成情報をもとに、リポジトリ管理部、HTML生成エンジンの制御を行い、HTML生成エンジンが、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報および切り張りWebページの構成情報の取り出しを行い、取り出された情報をもとに、HTMLファイルを作成するように構成される。

【0025】

【発明の実施の形態】実施の形態1. 本発明の実施の形態に付いて図1を用いて説明する。図1は本発明のWebページ生成装置の一構成を示す図である。図1において、1は複数のWebページからオブジェクトを抽出し、自由なレイアウトで新たなWebページを作成するWebページ作成部、2は作成されたWebページの構成情報を格納すると共に、貼り付けられたオブジェクトを更新するリポジトリ管理部、3はリポジトリ管理部に格納された構成情報とオブジェクトから実際にWebページを作成するWebページ実行部、20はWebページ実行部3で生成された切り張りWebページをネットワークにシェアするためのWebサーバ、21は切り張りWebページを参照するためのWebブラウザ21である。

【0026】図2はWebページ作成部1の構成を示す

図である。図2において、Webページ作成部1はオブジェクトを抽出するオブジェクト抽出エディタ4、Webページ作成エディタ5から構成される。さらに、オブジェクト抽出エディタ4は、HTML解析部50とオブジェクト修正部51とから構成される。オブジェクト修正部51は、HTML解析エンジン11および解析ルールDB12から構成され、オブジェクト修正部51はオブジェクトパッケージ部13から構成される。Webページ作成エディタ5はオブジェクト表示部14、リポジトリI/O部15およびWebページレイアウト部16から構成される。

【0027】図3はWebページ実行部3の構成を示す図である。図3において、Webページ実行部3は、Webページリレイアウト部9と実行制御部10とから構成される。Webページリレイアウト部9は、リポジトリI/O部52とHTML生成エンジン18とから構成される。実行制御部10は、実行マネージャ19とリポジトリI/O部53から構成される。

【0028】図4は、リポジトリ管理部2の構成を示す図である。図4において、リポジトリ管理部2は、オブジェクトチェックエンジン6、Webページ構成情報DB7およびオブジェクトDB8から構成される。さらに、オブジェクトチェックエンジン6は、リポジトリI/O部54、HTML解析エンジン11およびオブジェクト比較部17から構成される。

【0029】以下に、一例として、ある人が、〇×スキー場に行く場合、〇×地区の天気と〇×スキー場の宿の予約情報を調べるために、図5に示すようなインターネット上の天気情報のWebページと図6に示す宿予約状況のWebページを参照し、図5から天気図だけを抜き出し、図6から予約情報だけを抜き出して、これを図7に示すような「〇×スキー場総合情報」のような切り張り型Webページを新たに作成し、Webブラウザ21で参照する場合について、本発明のWebページ生成装置における各部の動作について以下に説明する。

【0030】図8は、Webページ作成部1でオブジェクトを抽出し、その抽出したオブジェクトからWebページを作成する最も基本的なフローチャートを示す図である。最初にWebページ作成時の動作について図8に示す動作フローを説明する。Webページ作成においては、図8に示すように、まず、ステップS20で、ユーザはオブジェクトを抽出し、次に、ステップS21で、その抽出したオブジェクトを加工してWebページを作成する。図9は、図8のステップS20の詳細、すなわち、オブジェクト抽出エディタ4の動作を説明するフローチャートを示す図である。図10は、図8のステップS21の詳細、すなわち、Webページ作成エディタ5の動作を説明するフローチャートを示す図である。

【0031】オブジェクト抽出

図9において、オブジェクト抽出エディタ4は、ステッ

プS22において、図5と図6に示される2つのWebページを読み込み、ステップS23において、Webページをオブジェクトに分割する。この分割は、HTML解析エンジン11が解析ルールDB12のルールに従って行う。図16は、解析ルールDB12のテーブル内容を示す図である。HTML解析エンジン11は、図16の解析ルールDBに格納された解析ルールに従い、

「図」の部分は図17の231を分割し、「表」の部分は図18の233をオブジェクトとして分割する。ここで、図17および図18は、それぞれ天気情報と宿情報を提供するオリジナルの図5と図6のWebページに対するソースを示す図である。

【0032】図16において、解析ルールDB12は、オブジェクトタイプ、先頭アンカータグ、終端アンカータグから構成される。図16の解析ルールDB12においては、図16のオブジェクトタイプ223の場合には、先頭アンカータグは、たとえば、「<p><image src =」224がストアされ、終端アンカータグ部は、たとえば、「></p>」225がストアされる。一方、表のオブジェクトタイプ226の場合には、先頭アンカータグは、たとえば、「<table」227がストアされ、終端アンカータグ部は、たとえば、「<table>」228がストアされる。上記の「<p><image src =」224および「></p>」225は、図17の231に対応し、「<table」227および「<table>」228は、図18の233中の235および234に対応する。

【0033】ステップS24において、分割されたオブジェクトはオブジェクトパッケージ部13に渡される。ステップS25において、ユーザは、オブジェクトパッケージ部13において、天気図に関しては、図17の「〇×地方の天気図」230の記述を一つのオブジェクトとして修正し、同様に、宿予約表に関しては図18の「宿の予約状況」232の記述を一つのオブジェクトとして修正する。ステップS26において、ユーザは分割の仕方が正しいかをチェックし、もしその分割の仕方が正しければ、Webページは、図10に示されるWebページ作成エディタ5に渡される。

【0034】Webページ作成エディット

図10のステップS27において、Webページ作成エディタ5は、オブジェクト抽出エディタ4で抽出・修正されたオブジェクトを取り込む。ステップS28において、Webページ作成エディタ5は、リポジトリI/O部15を介してリポジトリ管理部2に既に格納されているオブジェクトを利用するか否かを判断する。もし、リポジトリ管理部2に既に格納されているオブジェクトを利用する場合は、リポジトリ管理部2に格納されたオブジェクトが取り出され、オブジェクト表示部14に渡される。ステップS30において、オブジェクト表示部14はリポジトリI/O部15を介してリポジトリ管理部2から受けたオブジェクトを表示する。ステップS31

において、ユーザはWebページレイアウト部16で、オブジェクト表示部に表示されたオブジェクトをエディタ上に貼り付けて、図7の「〇×スキー場総合情報」のWebページのレイアウトを行う。ステップS34において、レイアウトが完了すると、Webページの構成情報とオブジェクト情報がリポジトリ管理部2に格納される。

【0035】オブジェクトDBの構成

リポジトリ管理部2のオブジェクトDB8の構成は図14に示される。図14に示されるオブジェクトDB8は、オブジェクトID、オリジナルURL、先頭アンカータグ、終端アンカータグ、オブジェクトタイプ、オブジェクト本体、最終更新日時、更新頻度の各フィールドから構成される。たとえば、図7の天気図のオブジェクトは、オブジェクトDB8中に、それぞれオブジェクトIDが「A000001」204、オリジナルURLが「http://ab/c.htm」205、先頭アンカータグが「<p 〇×地方の天気図</p>」206、終端アンカータグが「<p><img src=」207、オブジェクトタイプが「図」208、オブジェクト本体が「天気図データ(JPG)」209、最終更新日時が「1997.11.1.13:00」210、更新頻度が「毎日、13:00」219のように格納される。

【0036】一方、表のオブジェクトは、オブジェクトDB8中に、それぞれオブジェクトIDが「A00000」211、オリジナルURLが「http://tr/f.htm」212、先頭アンカータグが「<p><big><big>宿の予約状況」213、終端アンカータグが「</table」214、オブジェクトタイプが「表」215、オブジェクト本体が「表データ(HTML)」216、最終更新日時が「1997.11.1.12:00」217、更新頻度が「毎日、0:00,12:00」218のように格納される。このように、図14の「先頭アンカータグ」206、213にはオブジェクト抽出エディタ4でユーザが図および表のオブジェクトに文字列を含めた修正が反映されている。また、Webページ作成エディタで貼り付けたそれぞれのオブジェクトの更新頻度が定義でき、これが図14のオブジェクトDB8の更新頻度219、218に登録されていることが分かる。

【0037】Webページ構成情報DBの構成

一方、リポジトリ管理部2のWebページ構成情報DB7の構成は図15に示される。属性値15のWebページ構成情報DB7は、切り張りWebページURL名、切り張りWebページレイアウト情報(HTML形式)、更新フラグから構成される。たとえば、図7の表のオブジェクトは、Webページ構成情報DB7中に、切り張りWebページURL名が「http://MyServe/ski.htm」220、切り張りWebページレイアウト情報(HTML形式)が「レイアウト」221、更新フラグが「ON」のように格納される。Webページ作成エディタ5では作成したWebページのオブジェクトの更新情報(参照時

に更新するかどうか)の設定も行う。ここではユーザが「参照時に更新」したので、図15の更新フラグ222はONになっている。

【0038】図19は、総合情報Webページのレイアウト情報を示す図である。図19には、図15中の切り張りWebページレイアウト情報(HTML形式)のフィールド221に格納された「レイアウト情報」を示す。図19の238および239は、図14のオブジェクトIDフィールド201および211に記述された「<C_P_ObjectID=A000001>」、239および「<C_P_ObjectID=A000002>」と対応している。

【0039】Webページの参照

以下に、Webページを参照する動作について、図11、図12および図13に示す動作フローチャートに基づいて説明する。図11は、実施例1における切り張りWebページ参照時の動作を説明するフローチャートを示す図である。図12は、図11中のWebページ実行部3の動作を説明するフローチャートを示す図である。

【0040】図11のステップS35において、ユーザはWebブラウザ21に「〇×総合スキー情報」のURLを入力する。ステップS36において、Webサーバ20はWebブラウザ21からの依頼を受け、Webページ実行部3にURLを渡す。

【0041】Webページ実行部の動作

実行制御部10がリポジトリ管理部2中の切り張りWebページ構成情報DB8を検索し、図15のフィールド220に示されるURL名に対応したWebページの構成情報を取り出す。Webページ実行部3は、図12に詳細に示すように、ステップS38において、実行制御部10がリポジトリ管理部2中の切り張りWebページ構成情報DB8を検索し、図15のフィールド220に示されるURL名に対応したWebページの構成情報を取り出す。ステップS38において、Webページの構成情報中の更新フラグフィールド222が「ON」であるか否かが判断される。更新フラグフィールド222が「ON」であると、処理はステップS41に移る。ステップS41において、実行制御部10はWebページの構成情報中のレイアウト情報(図19)に存在するオブジェクトID(「A000001」(図19の238), 「A000002」(図19の239))の更新をリポジトリI/O部53を介してリポジトリ管理部2に依頼する。

【0042】一方、ステップS38において、更新フラグフィールド222が「ON」でない場合は、実行制御部10はWebページレイアウト部9に対して切り張りWebページ構成情報を渡し、Webページの作成を依頼する。ステップS40において、Webページレイアウト部9は渡された切り張りWebページ構成情報上にあるオブジェクトをリポジトリI/O部52を介してリポジトリ管理部2から取り出し、Webページを

作成する。ステップS40で作成されたWebページは図11のステップS42において、Webサーバ20を通じてWebブラウザ21に表示される。

【0043】リポジトリ管理部の動作

図13は、図11中のリポジトリ管理部2の動作を説明するフローチャートを示す図である。リポジトリ管理部2は、図12のステップS41における実行制御部10からオブジェクトID("A000001"(図19の238), "A000002"(図19の239))の更新依頼されると、ステップS43において、オブジェクトチェックエンジン6がオブジェクトIDをキーにオブジェクトDB8を検索し、オブジェクトの属性情報であるオリジナルURL、およびオブジェクトを抽出するためのルール(先頭・終端アンカータグ)を得る。ステップS44において、オブジェクト比較部17は作成されたWebページとオリジナルURLで示されるWebページとを接続する。ステップS45において、オリジナルWebページの作成日時がオブジェクトDB8の最終更新日時(図14のフィールド210)より新しいか否かが判断される。もし、オリジナルWebページの作成日時がオブジェクトDB8の最終更新日時より新しければ、ステップS46において、HTML解析エンジン11はルール(先頭・終端アンカータグ)に基づいて、オリジナルURLからオブジェクトを取り出す。ステップS47において、取り出されたオブジェクトはリポジトリI/O部54によりオブジェクトDB8に再登録される。ステップS48において、オブジェクトチェックエンジン6はWebページ実行部3に対してチェック終了を通知する。また、ステップS45において、もし、オリジナルWebページの作成日時がオブジェクトDB8の最終更新日時より新しくなければ、処理はステップS48に移り、オブジェクトチェックエンジン6はWebページ実行部3に対してチェック終了を通知する。

【0044】図12に戻って、Webページ実行部3がリポジトリ管理部2からチェック終了通知を受領すると、ステップS39において、実行制御部10はHTML生成エンジン18に先にリポジトリ管理部から取り出したWebページレイアウト情報(図19)を渡し、Webページの生成を依頼する。ステップS40において、HTML生成エンジン18はWebページレイアウト情報のオブジェクトID部(図19のデータ238, 239)をとオブジェクトDB8から取り出した更新オブジェクトに入れ替えて、図7のWebページ(図20のHTMLソース)を作成する。ステップS40で作成されたWebページは図11のステップS42において、Webサーバ20を通じてWebブラウザ21に表示される。

【0045】

【発明の効果】第1と第12の発明によれば、本発明は、Webページ作成部が、複数のWebページからオ

ブジェクトを抽出し、自由なレイアウトで新たなWebページを作成し、リポジトリ管理部が、作成されたWebページの構成情報を格納すると共に、貼り付けられたオブジェクトを更新し、Webページ実行部が、前記リポジトリ管理部に格納された構成情報とオブジェクトから実際にWebページを作成するように構成されるので、複数のWebページの中から必要なオブジェクトを切り出し、自分の好みにレイアウトし直して、新たなWebページが作成できる。さらに、作成したWebページ上に配置されたオブジェクトはオリジナルのWebページ上のオブジェクトが更新されれば、新たなWebページは自動的に最新のものに更新される。従って、インターネット上に分散している情報を自由なレイアウトで、しかも参照時に最新の情報を見ることが可能となる効果がある。

【0046】第2と第13の発明によれば、本発明は、Webページ作成部では、オブジェクト抽出エディタが、既存の複数のWebページからオブジェクトを抽出し、Webページ作成エディタが、抽出されたオブジェクトをエディットするように構成されるので、既存のWebページから新たなWebページを容易に作成することができる。

【0047】第3と第14の発明によれば、本発明は、オブジェクト抽出エディタでは、HTML解析部が、入力されたWebページをオブジェクト毎に分割し、オブジェクト修正部が、前記HTML解析部で分割されたオブジェクトの分割の仕方を修正することによって、既存のWebページからオブジェクトを抽出するように構成されるので、既存のWebページから大まかにオブジェクトを抽出し、ユーザの指示のもと、細かなオブジェクトの抽出の修正ができる。従って、既存のWebページから効率よくオブジェクトの抽出を行うことができる。

【0048】第4と第15の発明によれば、本発明は、HTML解析部では、解析ルールDBが、オブジェクトをWebページから抽出するためのルールを格納し、HTML解析エンジンが、前記解析ルールDBに格納されたルールに従って、入力されたWebページを与えられたオブジェクトに分割する構成されるので、既存のWebページ上からのオブジェクト抽出修正作業がマウスなどの感覚的な操作により行えるため、ユーザのオブジェクト抽出修正作業の効率化がはかれる。また、オブジェクトの持つ属性情報も付加することができる。

【0049】第5と第16の発明によれば、本発明は、前記オブジェクト修正部では、オブジェクトパッケージ部が、前記HTML解析部によってオブジェクトに分割されたWebページをユーザに表示しオブジェクトの分割の修正を行い、およびオブジェクトの持つ属性情報を付加するように構成されるので、今までに無い新たなタイプのオブジェクトを抽出する場合でも、解析ルールDBに抽出ルールを追加することにより対応できる

【0050】第6と第17の発明によれば、本発明は、Webページ作成エディタでは、オブジェクト表示部がリポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報や作成した切り張りWebページの構成情報の登録および取り出し等を行い、抽出されたオブジェクトをグラフィカルに表示してユーザにWebページ作成のための部品として抽出されたオブジェクトを提供し、Webページレイアウト部がオブジェクト表示部に表示されたオブジェクトを配置して切り張りWebページを作成するように構成されるので、複数のWeb

ページから抽出されたオブジェクトをマウスで配置して、新たなWebページが作成できるので、Webページの作成の操作性を向上させる。

【0051】第7と第18の発明によれば、本発明は、リポジトリ管理部では、オブジェクトDBが、抽出したWebページ上のオブジェクトおよび、オブジェクトの属性情報を管理し、切り張りWebページ構成情報DBが、作成したWebページのレイアウト情報を管理し、オブジェクトチェックエンジンが、オブジェクトの更新を行うように構成されるので、Webページのレイアウトとその上のオブジェクトが独立して管理できる。これにより、ユーザは1つのオブジェクトを複数の切り張りWebページに利用でき、そのオブジェクト間の整合性などを気にする必要がない。また、更新フラグの情報により、ページの生成の方法（オブジェクトの更新をおこなってから、生成する等）がページ単位で設定できる。また、分散して存在するWebページ上のオブジェクトの管理が一元化できるため、更新のチェックなどが容易になり、また、それを利用したWebページ再利用性も向上する。

【0052】第8と第19の発明によれば、本発明は、オブジェクトチェックエンジンでは、HTML解析エンジンが、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報の登録、取り出しを行い、取り出したオブジェクト情報の中の抽出ルールに従って、オブジェクトを解析し抽出し、オブジェクト比較部が、リポジトリI/O部から渡されるリポジトリ管理部に登録されているオブジェクトおよびHTML解析エンジンから渡されるオリジナルのオブジェクトを比較し、オブジェクトが更新されていればリポジトリI/O部を介してリポジトリ管理部にオブジェクトの再登録を依頼するように構成されるので、Webページ上のオブジェクトの更新チェック、切り出し、再登録が自動的に行える。従って、ユーザは意識すること無く、常に最新情報を参照することができる。また、更新をバックグラウンドでおこなうことにより、参照時にオブジェクトのチェックを行わずに切り張りWebページを作成できるので、参照時のレスポンス向上にもつながる。

【0053】第9と第20の発明によれば、本発明のWebページ実行部は、リポジトリ管理部からURL名に

対応したWebページの構成情報を取り出す実行制御部と、前記実行制御部からの切り張りWebページ構成情報に基づいてWebページの作成を行うWebページレイアウト部とを備えるように構成されるので、切り張りWebページのリポジトリから、Webページを動的に作成することが可能となる。

【0054】第10の発明によれば、本発明のWebページ実行部は、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報および切り張りWebページの構成情報の取り出しを行い、取り出されたオブジェクトの情報および切り張りWebページのレイアウト情報をもとに、HTMLファイルを作成するHTML生成エンジンを備えるように構成されるので、切り張りWebページのリポジトリから、Webページを動的に作成することが可能となる。従って、Webページの更新などの作業をユーザが気にする必要なくなる。

【0055】第11の発明によれば、本発明の実行制御部は、リポジトリI/O部を介してリポジトリ管理部との間でオブジェクトの情報および切り張りWebページの構成情報の取り出しを行い、取り出されたオブジェクトの情報および切り張りWebページの構成情報をもとに、リポジトリ管理部およびWebページレイアウト部の制御を行う実行マネージャとを備え、Webページの生成を制御するように構成されるので、切り張りWebページの性質（定期的に更新、参照時に更新）等により切り張りWebページの生成手順が変更でき、効率的な参照が可能となる。また、分散して存在するWebページ上のオブジェクトの管理が一元化でき、再利用性も向上する。

30 【図面の簡単な説明】

【図1】 本発明の実施の形態における切り張り型Webページ生成装置の全体ブロックを示す図である。

【図2】 本発明の実施の形態におけるWebページ作成部の構成を示す図である。

【図3】 本発明の実施の形態におけるWebページ実行部の構成を示す図である。

【図4】 本発明の実施の形態におけるリポジトリ管理部の構成を示す図である。

【図5】 本発明の実施の形態における天気図のオリジナルWebページを示す図である。

【図6】 本発明の実施の形態における宿情報のオリジナルWebページを示す図である。

【図7】 本発明の実施の形態における生成された切り張り型Webページ（総合情報Webページ）を示す図である。

【図8】 本発明の実施の形態における切り張りWebページ生成時の動作フローチャートを示す図である。

【図9】 本発明の実施の形態におけるオブジェクト抽出エディタの動作を説明するフローチャートを示す図である。

17

【図10】 本発明の実施の形態におけるWebページ作成エディタの動作を説明するフローチャートを示す図である。

【図11】 本発明の実施の形態における切り張りWebページ参照時の動作を説明するフローチャートを示す図である。

【図12】 本発明の実施の形態におけるWebページ実行部の動作を説明するフローチャートを示す図である。

【図13】 本発明の実施の形態におけるリポジトリ管理部の動作を説明するフローチャートを示す図である。

【図14】 本発明の実施の形態におけるオブジェクトDBのテーブル内容を示す図である。

【図15】 本発明の実施の形態における切り張りWebページ構成情報DBのテーブル内容を示す図である。

【図16】 本発明の実施の形態における解析ルールDBのテーブル内容を示す図である。

【図17】 本発明の実施の形態における天気図のオリジナルWebページのソースを示す図である。

【図18】 本発明の実施の形態における宿情報のオリジナルWebページのソースを示す図である。

【図19】 本発明の実施の形態における総合情報Web

18

bページのレイアウト情報を示す図である。

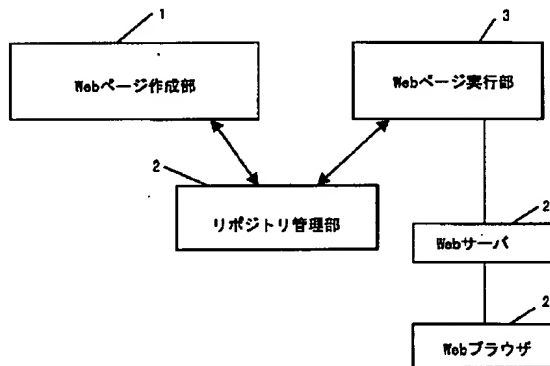
【図20】 本発明の実施の形態における生成された総合情報Webページのソースを示す図である。

【図21】 従来のリンク集を用いたWebページの参照方法を示す図である。

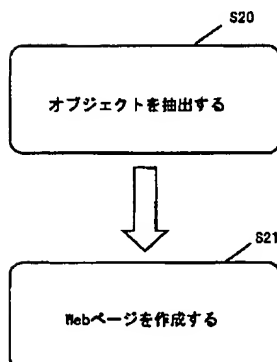
【符号の説明】

1…Webページ作成部、2…リポジトリ管理部、3…Webページ実行部、4…オブジェクト抽出エディタ、5…Webページ作成エディタ、6…オブジェクトチェックエンジン、7…Webページ構成情報DB、8…オブジェクトDB、9…Webページリレイアウト部、10…実行制御部、11…HTML解析エンジン、12…解析ルールDB、13…オブジェクトパッケージ部、14…オブジェクト表示部、15…リポジトリI/O部、16…Webページレイアウト部、17…オブジェクト比較部、18…HTML生成エンジン、19…実行マネージャ、20…Webサーバ、21…Webブラウザ、50…HTML解析部、51…オブジェクト修正部、52…リポジトリI/O部、53…リポジトリI/O部、54…リポジトリI/O部

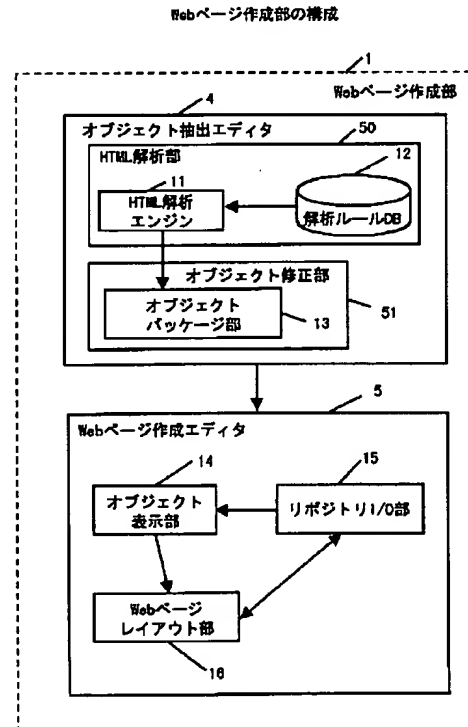
【図1】



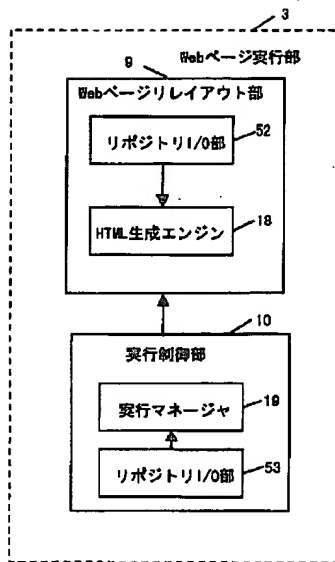
【図8】



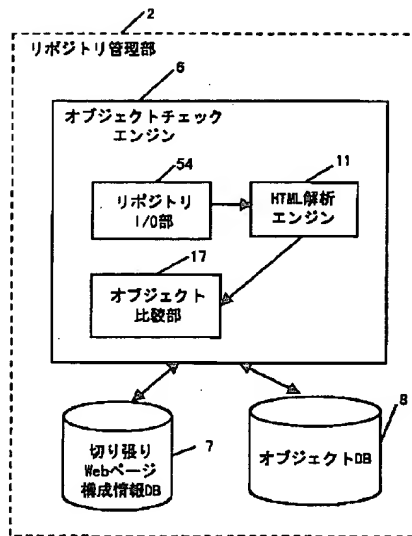
【図2】



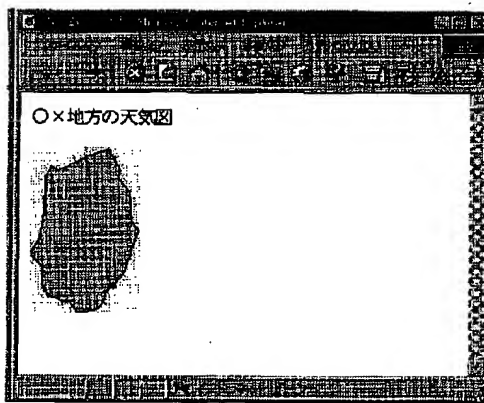
【図3】



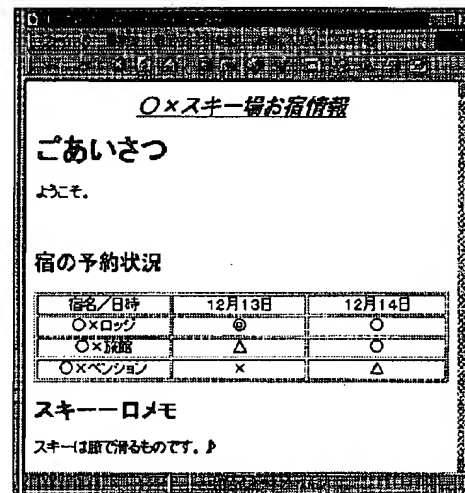
【図4】



【図5】



【図6】

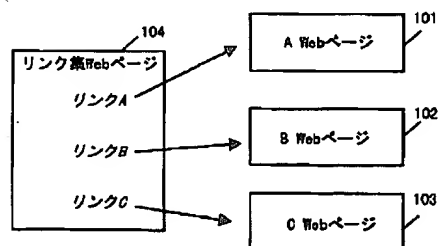


【図15】

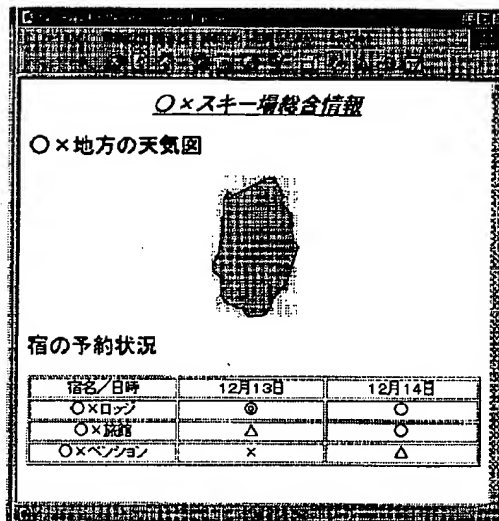
切り張りWebページ構成情報DBの内容

切り張りWeb ページURL名	Webページレイアウト 情報 (HTML形式)	更新フラグ
http://MyServer/s ki.htm	レイアウト情報	ON

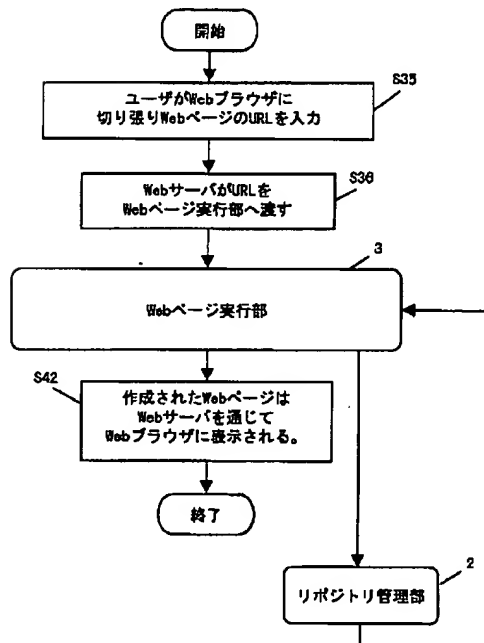
【図21】



【図7】

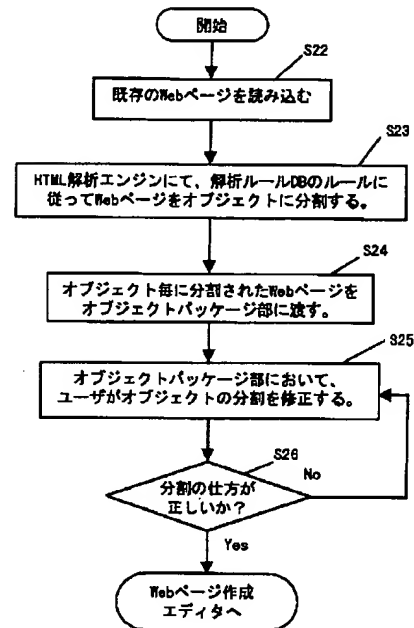


【図11】



【図9】

オブジェクト抽出エディタの動作を示すフローチャート



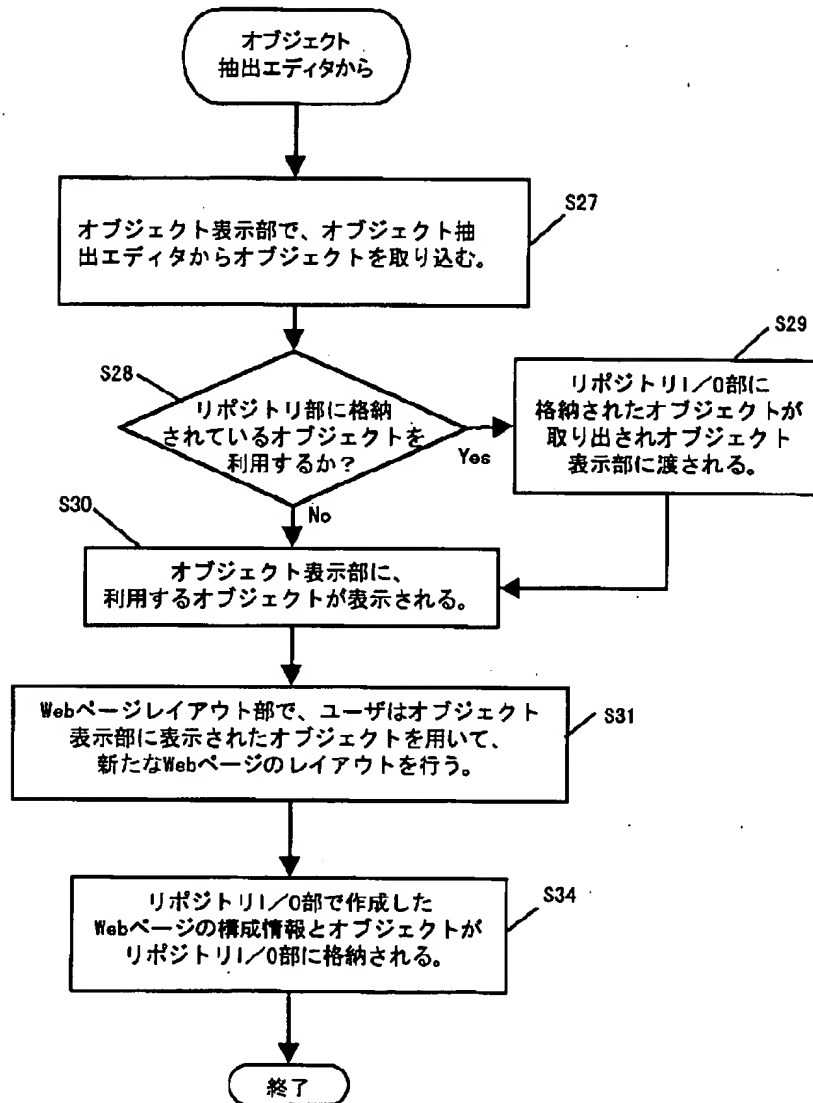
【図16】

解析ルールDBのテーブル構成

オブジェクトタイプ	先頭アンカータグ	終端アンカータグ
図	<p>>/p>
表	<table	<table>

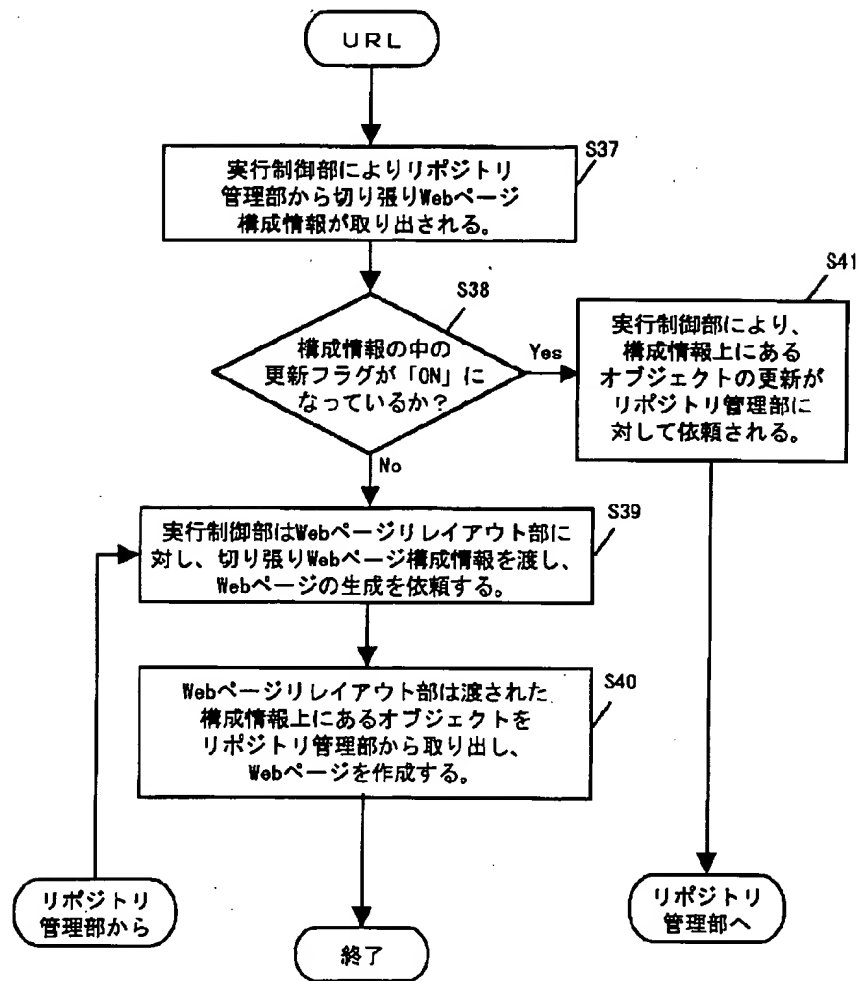
【図10】

Webページ作成エディタの動作を示すフローチャート



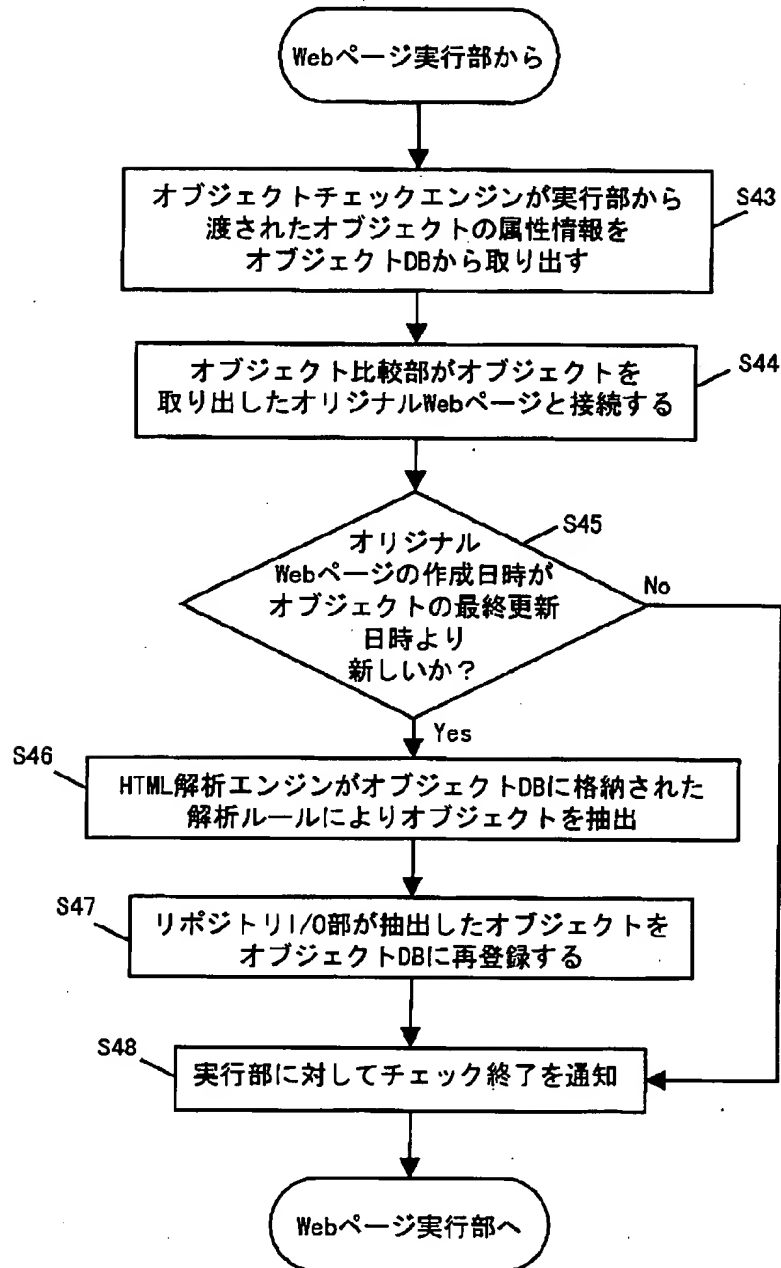
【図12】

Webページ実行部の動作を示すフローチャート



【図13】

リポジトリ管理部の動作を示すフローチャート



【図14】

オブジェクトDBのテーブル内容

オブジェクトID	オリジナルURL	先頭アンカー タグ	終端アンカー タグ	オブジェクト タイプ	オブジェクト 本体	最終更新日時	更新頻度
A000001	http://ab/o.htm	<p>〇×地方の 天気図</p>	<p>				
<title>ホーム ページ</title>
</head>

<body>

<p>〇×地方の天気図</p>

<p></p>

<p> </p>
</body>
</html>

```

【図18】

図6のHTMLソース

```

<html>
<head>
<<中略>>

<p align="center"><big><big>〇×スキー場お宿情報</big></p>
<p><big><big>ごあいさつ</big></p>
<p>ようこそ.</p>
<p> </p>
<p><big><big>宿の予約状況</big></p>
<div align="center"><center>
<table border="1" width="100%">
<tr>
<td width="33%" align="center">宿名/日時</td>
<td width="33%" align="center">1 2月13日</td>
<td width="34%" align="center">1 2月14日</td>
</tr>
<tr>
<td align="center">〇×ロッジ</td>
<td align="center">△</td>
<td align="center">〇</td>
</tr>
<tr>
<td align="center">〇×旅館</td>
<td align="center">△</td>
<td align="center">〇</td>
</tr>
<tr>
<td align="center">〇×ペンション</td>
<td align="center">×</td>
<td align="center">△</td>
</tr>
</table>
</center></div>
<p><big><big>スキー—ロモ</big></p>
<p>スキーは陸で滑るものです.</p>
</body>
</html>

```

【図20】

図7のHTMLソース

```

<html>
<head>
<<中略>>

<body>
<p align="center"><big><big>〇×スキー場総合情報</big></p>
<p><big><big>宿の予約状況</big></p>
<div align="center"><center>
<table border="1" width="100%">
<tr>
<td width="33%" align="center">宿名/日時</td>
<td width="33%" align="center">1 2月13日</td>
<td width="34%" align="center">1 2月14日</td>
</tr>
<tr>
<td align="center">〇×ロッジ</td>
<td align="center">△</td>
<td align="center">〇</td>
</tr>
<tr>
<td align="center">〇×旅館</td>
<td align="center">△</td>
<td align="center">〇</td>
</tr>
<tr>
<td align="center">〇×ペンション</td>
<td align="center">×</td>
<td align="center">△</td>
</tr>
</table>
</center></div>
</body>

```

【図19】

図7のHTMLソース

```
<html>
<head>
 <<中路>>
<body>
<p align="center"><u><big><big>〇スキー場総合情報
<p align="center"></p>
<div align="center"><center>
<<C_P_ObjectID=A000002>
</center></div>
</body>
</html>
```

フロントページの続き

(51)Int. Cl.<sup>6</sup>

識別記号

FI

G06F 15/62

325A

PTO 03-3009

CY=JA DATE=19990917 KIND=A  
PN=11-250054

MERGING WEB PAGE GENERATION DEVICE AND METHOD THEREFOR  
[Kirihari-gata Web peeji seisei sochi oyobi seisei hoho]

Tetsuya Kosaka, et al.

UNITED STATES PATENT AND TRADEMARK OFFICE  
Washington, D.C. May 2003

Translated by: FLS, Inc.

○

PUBLICATION COUNTRY (10): JP

DOCUMENT NUMBER (11): 11250054

DOCUMENT KIND (12): A

PUBLICATION DATE (43): 19990917

PUBLICATION DATE (45):

APPLICATION NUMBER (21): 10049508

APPLICATION DATE (22): 19980302

ADDITION TO (61):

INTERNATIONAL CLASSIFICATION (51): G06F 17/24; G06F 17/21; G06F 17/30; //G06T 11/60

DOMESTIC CLASSIFICATION (52):

PRIORITY COUNTRY (33):

PRIORITY NUMBER (31):

PRIORITY DATE (32):

INVENTOR (72): KOSAKA, TETSUYA; FUJIMOTO, MAYUMI; WATABE, NAOSUKE

APPLICANT (71): MITSUBISHI ELECTRIC CORPORATION

TITLE (54): MERGING WEB PAGE GENERATION DEVICE AND METHOD THEREFOR

FOREIGN TITLE [54A]: KIRIHARI-GATA WEB PEEJI SEISEI SOCHI OYOBI SEISEI HOHO

SPECIFICATIONS

/1

[Title of the Invention]

Merging Web Page Generation Device and Method Therefor

[Claims]

/2

[Claim 1] Web page generation device, being a web page generation device for generating and viewing a new Web page using several existing Web pages, characterized by having

a Web page generation part for extracting objects from several Web pages and generating a new Web page in a chosen layout,

a repository management part for storing the configuration information of the generated Web page and updating affixed objects, and

a Web page execution part for generating an actual Web page from the configuration information and objects stored in the above-mentioned repository management part.

[Claim 2] Web page generation device, characterized by the above-mentioned Web page generation part in the Web page generation device described in Claim 1 having

an object extraction editor for extracting objects from several existing Web pages, and

a Web page generation editor for editing extracted objects.

[Claim 3] Web page generation device, characterized by the above-mentioned object extraction editor in the Web page generation device described in Claim 2 having

an HTML analysis part having a function for dividing an inputted Web page into objects and

an object revision part having a function for revising how objects divided by the above-mentioned HTML analysis part have been divided, and

extracting objects from existing Web pages.

---

\*Numbers in the margin indicate pagination in the foreign text.

[Claim 4] Web page generation device, characterized by the above-mentioned HTML analysis part in the Web page generation device described in Claim 3 having

an analysis rules DB for storing rules for extracting objects from Web pages,

an HTML analysis engine for dividing an inputted Web page into given objects according to the rules stored in the above-mentioned analysis rules DB, and

an analysis rules DB where the rules for extracting objects from Web pages have been stored.

[Claim 5] Web page generation device, characterized by the above-mentioned object revision part in the Web page generation device described in Claim 3 having

an object package part for displaying Web pages divided into objects by the above-mentioned HTML analysis part to the user and having an editor function for revising how objects have been divided and a function for appending attribute information to objects.

[Claim 6] Web page generation device, characterized by the above-mentioned Web page generation editor in the Web page generation device described in Claim 1 having

an object display part for recording and reading object information and the configuration information of the generated merging Web page to and from the repository management part through a repository I/O part and offering extracted objects to the user as parts for generating a Web page by displaying extracted objects graphically, and

a Web page layout part for generating a merging Web page by laying out objects displayed by the above-mentioned object display part.

[Claim 7] Web page generation device, characterized by the above-mentioned repository management part in the Web page generation device described in Claim 1 having

an object DB for managing objects on an extracted Web page and object attribute information,

a merging Web page configuration information DB for managing the layout information of a generated Web page, and

an object check engine for updating objects.

[Claim 8] Web page generation device, characterized by the above-mentioned object check engine in the Web page generation device described in Claim 7 having

an HTML analysis engine for recording and reading object information to and from the repository management part through a repository I/O part and analyzing, extracting, and sending objects to the object comparison part according to the extraction rules in the read object information, and

an object comparison part for comparing objects recorded in the repository management part sent from the repository I/O part to original objects sent from the HTML analysis engine and requesting the repository I/O part to rerecord an object in the repository management part if the object has been updated.

[Claim 9] Web page generation device, characterized by the above-mentioned Web page execution part in the Web page generation device described in Claim 1 having

an execution control part for reading the configuration information of Web pages corresponding to URL names from the repository management part, and

a Web page re-layout part for generating a Web page based on the merging Web page configuration information from the above-mentioned execution control part.

[Claim 10] Web page generation device, characterized by the above-mentioned Web page re-layout part in the Web page generation device described in Claim 9 having

an HTML generation engine part for recording and reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and generating an HTML file based on the read object information and Web page layout information.

[Claim 11] Web page generation device, characterized by the above-mentioned execution control part of the Web page generation device described in Claim 9 having

an execution manager for reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and



controlling the repository management part and the Web page layout part based on the read object information and merging Web page configuration information, and

controlling generation of Web pages.

[Claim 12] Web page generation method, being a web page generation device for generating and viewing a new Web page using several existing Web pages, characterized by

a Web page generation part extracting objects from several Web pages and generating a new Web page in a chosen layout,

a repository management part storing the configuration information of the generated Web page and updating affixed objects, and

a Web page execution part generating an actual Web page from the configuration information and objects stored in the above-mentioned repository management part.

[Claim 13] Web page generation method, wherein the above-mentioned Web page generation part in the Web page generation device described in Claim 12 is characterized by

an object extraction editor extracting objects from several existing Web pages, and

a Web page generation editor editing extracted objects.

[Claim 14] Web page generation method, wherein the above-mentioned object extraction editor in the Web page generation device described in Claim 13 is characterized by

an HTML analysis part dividing an inputted Web page into objects,

an object revision part revising how objects divided by the above-mentioned HTML analysis part have been divided, and

extracting objects from existing Web pages.

[Claim 15] Web page generation method, wherein the above-mentioned HTML analysis part in the Web page generation device described in Claim 14 is characterized by

an analysis rules DB storing rules for extracting objects from

Web pages, and

an HTML analysis engine dividing an inputted Web page into given objects according to the rules stored in the above-mentioned analysis rules DB.

[Claim 16] Web page generation method, wherein the above-mentioned object revision part in the Web page generation device described in Claim 14 is characterized by

an object package part displaying Web pages divided into objects by the above-mentioned HTML analysis part to the user, revising how objects have been divided, and appending attribute information to objects.

[Claim 17] Web page generation method, wherein the above-mentioned Web page generation editor in the Web page generation device described in Claim 13 is characterized by

an object display part recording and reading object information and the configuration information of the generated merging Web page to and from the repository management part through a repository I/O part and offering extracted objects to the user as parts for generating a Web page by displaying extracted objects graphically, and a Web page layout part generating a merging Web page by laying out objects displayed by the above-mentioned object display part.

[Claim 18] Web page generation method, wherein the above-mentioned repository management part in the Web page generation device described in Claim 12 is characterized by

managing objects on an extracted Web page and object attribute information in an object DB,

managing the layout information of a generated Web page in a merging Web page configuration information DB, and processing such as updating objects by an object check engine.

[Claim 19] Web page generation method, wherein the above-mentioned object check engine in the Web page generation device described in Claim 18 is characterized by

an HTML analysis engine recording and reading object information to and from the repository management part through a repository I/O part and analyzing, extracting, and sending objects to the object comparison part according to the extraction rules in the read object information, and an object comparison part comparing objects recorded

in the repository management part sent from the repository I/O part to original objects sent from the HTML analysis engine and requesting the repository I/O part to rerecord an object in the repository management part if the object has been updated.

[Claim 20] Web page generation method, wherein the above-mentioned Web page execution part in the Web page generation device described in Claim 12 is characterized by

an execution manager for reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and controlling the repository management part and the Web page layout part based on the read object information and merging Web page configuration information,

the above-mentioned HTML generation engine part recording and reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and generating an HTML file based on the read object information and Web page layout information.

[Detailed Explanation of the Invention]

[Industrial Field of Application]

This invention pertains to devices and methods for reusing objects (such as figures, tables, applets, or text strings) in Web pages offered on the Internet or the Intranet to generate and execute new Web pages.

/4

[Prior Art]

A method in prior art called "link assembly" is used for generating Web pages as shown in Figure 21 when associating independent information scattered on the Internet or the Intranet based on a given purpose. This method links the URL information of scattered Web pages (101), (102), and (103) together by generating link assembly Web page (104) describing each of these as a hyper-link. The user can display scattered Web pages (101), (102), and (103) on a Web browser by selecting the hyper-link tags on this link assembly Web page (104).

[Problems that the Invention is to Solve]

Normally, the unit of information that the user is searching is not one of the Web pages itself, but an object on a Web page. Linking

information in the manner of the "link assembly" by prior art, however, has the problem that not only is the user unable to view the information (objects) required in a chosen layout, but the user must open all of the linked Web pages on a browser to view the information required.

This invention was developed to solve problems such as described above. Its purpose is to make information (objects) required by the user and scattered on the Internet or the Intranet available for viewing in a layout preferred by the user.

#### [Means of Solving the Problems]

According to the first invention, this invention is comprised of a Web page generation part for extracting objects from several Web pages and generating a new Web page in a chosen layout, a repository management part for storing the configuration information of the generated Web page and updating affixed objects, and a Web page execution part for generating an actual Web page from the configuration information and objects stored in the above-mentioned repository management part.

According to the second invention, the Web page generation part of this invention is comprised of an object extraction editor for extracting objects from several existing Web pages, and a Web page generation editor for editing extracted objects.

According to the third invention, the object extraction editor of this invention is comprised of an HTML analysis part having a function for dividing an inputted Web page into objects and an object revision part having a function for revising how objects divided by the above-mentioned HTML analysis part have been divided, and extracts objects from existing Web pages.

According to the fourth invention, the HTML analysis part in this invention is comprised of an analysis rules DB for storing rules for extracting objects from Web pages, an HTML analysis engine for dividing an inputted Web page into given objects according to the rules stored in the above-mentioned analysis rules DB, and

an analysis rules DB where the rules for extracting objects from Web pages have been stored.

According to the fifth invention, the object revision part in this invention is comprised of an object package part for displaying Web pages divided into objects by the above-mentioned HTML analysis part to the user and having an editor function for revising how

objects have been divided and a function for appending attribute information to objects.

According to the sixth invention, the Web page generation editor in this invention is comprised of an object display part for recording and reading object information and the configuration information of the generated merging Web page to and from the repository management part through a repository I/O part and offering extracted objects to the user as parts for generating a Web page by displaying extracted objects graphically, and a Web page layout part for generating a merging Web page by laying out objects displayed by the above-mentioned object display part.

According to the seventh invention, the repository management part in this invention is comprised of an object DB for managing objects on an extracted Web page and object attribute information, a merging Web page configuration information DB for managing the layout information of a generated Web page, and an object check engine for updating objects.

According to the eighth invention, the object check engine in this invention is comprised of an HTML analysis engine for recording and reading object information to and from the repository management part through a repository I/O part and analyzing, extracting, and sending objects to the object comparison part according to the extraction rules in the read object information, and an object comparison part for comparing objects recorded in the repository management part sent from the repository I/O part to original objects sent from the HTML analysis engine and requesting the repository I/O part to rerecord an object in the repository management part if the object has been updated.

According to the ninth invention, the Web page execution part in this invention is comprised of an execution control part for reading the configuration information of Web pages corresponding to URL names from the repository management part, and a Web page re-layout part for generating a Web page based on the merging Web page configuration information from the above-mentioned execution control part.

According to the tenth invention, the Web page execution part in this invention is comprised of an HTML generation engine part for recording and reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and generating an HTML file based on the read object information and Web page layout information.

According to the eleventh invention, the execution control part in this invention is comprised of an execution manager for reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and controlling the repository management part and the Web page layout part based on the read object information and merging Web page configuration information, and controls generation of Web pages.

According to the twelfth invention, this invention is constituted by a Web page generation part extracting objects from several Web pages and generating a new Web page in a chosen layout, a repository management part storing the configuration information of the generated Web page and updating affixed objects, and a Web page execution part generating an actual Web page from the configuration information and objects stored in the above-mentioned repository management part.

According to the thirteenth invention, the Web page generation part of this invention is constituted by an object extraction editor extracting objects from several existing Web pages, and a Web page generation editor editing extracted objects.

According to the fourteenth invention, the object extraction editor of this invention is constituted by extracting objects from existing Web pages by an HTML analysis part dividing an inputted Web page into objects, and an object revision part revising how objects divided by the above-mentioned HTML analysis part have been divided.

According to the fifteenth invention, the HTML analysis part of this invention is constituted by an analysis rules DB storing rules for extracting objects from Web pages, and an HTML analysis engine dividing an inputted Web page into given objects according to the rules stored in the above-mentioned analysis rules DB.

According to the sixteenth invention, the object revision part of this invention is constituted by an object package part displaying Web pages divided into objects by the above-mentioned HTML analysis part to the user, revising how objects have been divided, and appending attribute information to objects.

According to the seventeenth invention, the Web page generation editor of this invention is constituted by an object display part recording and reading object information and the configuration information of the generated merging Web page to and from the repository management part through a repository I/O part and offering extracted objects to the user as parts for generating a Web page by displaying extracted objects graphically, and a Web page layout part

generating a merging Web page by laying out objects displayed by the above-mentioned object display part.

According to the eighteenth invention, this invention is constituted by the repository management part managing objects on an extracted Web page and object attribute information in an object DB, managing the layout information of a generated Web page in a merging Web page configuration information DB, and processing such as updating objects by an object check engine.

According to the nineteenth invention, the object check engine in this invention is constituted by an HTML analysis engine recording and reading object information to and from the repository management part through a repository I/O part and analyzing, extracting, and sending objects to the object comparison part according to the extraction rules in the read object information, and an object comparison part comparing objects recorded in the repository management part sent from the repository I/O part to original objects sent from the HTML analysis engine and requesting the repository I/O part to rerecord an object in the repository management part if the object has been updated.

According to the twentieth invention, the Web page execution part of this invention is constituted by an execution manager for reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and controlling the repository management part and the Web page layout part based on the read object information and merging Web page configuration information, the above-mentioned HTML generation engine part recording and reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and generating an HTML file based on the read object information and Web page layout information.

#### [Working Examples]

A working example of this invention will be explained using Figure 1. Figure 1 is a diagram showing one configuration of a Web page generation device of this invention. In Figure 1, (1) is a Web page generation part for extracting objects from several Web pages and generating a new Web page in a chosen layout, (2) is a repository management part for storing the configuration information of the generated Web page and updating affixed objects, (3) is a Web page execution part for generating an actual Web page from the configuration information and objects stored in the above-mentioned repository management part, (20) is a Web server for sharing merging

Web pages generated by Web page execution part (3) on a network, and (21) is a Web browser for viewing merging Web pages.

Figure 2 is a diagram showing the configuration of Web page generation part (1). Web page generation part (1) in Figure 2 is comprised of object extraction editor (4) for extracting objects, and Web page generation editor (5). Furthermore, object extraction editor (4) is comprised of HTML analysis part (50) and object revision part (51). Object revision part (51) [Translator's Note: error for HTML analysis part (50)] is comprised of HTML analysis engine (11) and analysis rules DB (12), and object revision part (51) is comprised of object package part (13). Web page generation editor (5) is comprised of object display part (14), repository I/O part (15), and Web page layout part (16).

/6

Figure 3 shows the configuration of Web page execution part (3). Web page execution part (3) in Figure 3 is comprised of Web page relayout part (9) and execution control part (10). Web page relayout part (9) is comprised of repository I/O part (52) and HTML generation engine (18). Execution control part (10) is comprised of execution manager (19) and repository I/O part (53).

Figure 4 shows the configuration of repository management part (2). Repository management part (2) in Figure 4 is comprised of object check engine (6), Web page configuration information DB (7), and object DB (8). Furthermore, object check engine (6) is comprised of repository I/O part (54), HTML analysis engine (11), and object comparison part (17).

Next, as an example, a certain person planning to go to OX ski area refers to a Web page on weather conditions shown in Figure 5 and a Web page on the status of lodging reservations shown in Figure 6 on the Internet to investigate the weather in OX region and lodgings in OX ski area. Extracting only the weather map from Figure 5 and only the reservation information from Figure 6, he or she generates a new merging Web page from this, such as "OX Ski Area Information" shown in Figure 7. The operation of each of the parts of the merging Web page generation device of this invention when viewing this on Web browser (21) will be explained below.

Figure 8 is a diagram showing the most basic flow chart for extracting objects by Web page generation part (1) and generating a Web page from these objects. The operational flow shown in Figure 8 will be explained for the operation when first generating a Web page. First, the user extracts an object in Step S20, then generates a Web page by adding this extracted object in Step S21. Figure 9 is a diagram showing the details of Step S20 in Figure 8; that is, a flow



chart illustrating the operation of object extraction editor (4). Figure 10 is a diagram showing the details of Step S21 in Figure 8; that is, a flow chart illustrating the operation of Web page generation editor (5).

### Extracting Objects

In Step S22, object extraction editor (4) in Figure 9 reads the two Web pages shown in Figures 5 and 6, and in Step S23, it divides the Web pages into objects. HTML analysis engine (11) analyzes these following the rules in analysis rules DB (12). Figure 16 is a diagram showing the table contents of analysis rules DB (12). Following the analysis rules stored in the analysis rules DB in Figure 16, HTML analysis engine (11) divides objects into "graphics" parts as (231) in Figure 17 and "table" parts as (233) in Figure 18. In this example, Figures 17 and 18 are diagrams showing the sources for the original Web pages in Figures 5 offering weather information and Figure 6 offering lodging information.

Analysis rules DB (12) in Figure 16 is comprised of the object type, header anchor tag, and footer anchor tag. In the case of object type (223) in Figure 16, a header anchor tag such as "<p><image src=" (224) and a footer anchor tag "></p>" such as (225) are stored in analysis rules DB in Figure 16. In the case of table object type (226), a header anchor tag such as "<table" (227) and a footer anchor tag such as "<table>" (228) are stored. The above-mentioned "<p><image src=" (224) and "></p>" (225) correspond to (231) in Figure 17, and the above-mentioned "<table" (227) and "<table>" (228) correspond to (235) and (234) within (233) in Figure 18.

In Step S24, the divided object is sent to object package part (13). In Step S25, the user revises the weather map in object package part (13) to one object called "Weather Map of OX Region", and likewise revises the lodging reservation table to one object called "Lodging Reservation Status" (232). In Step S26, the user checks whether the dividing method is correct. If the dividing method is correct, the Web page is sent to Web page generation editor (5) shown in Figure 10.

### Generating and Editing a Web Page

In Step S27 in Figure 10, Web page generation editor (5) reads the extracted and revised object in object extraction editor (4). In Step S28, Web page generation editor (5) determines through repository I/O part (15) whether or not an object already stored in repository management part (2) is used. If an object already stored in repository management part (2) is used, it reads and sends the

object stored in repository management part (2) to object display part (14). In Step S30, object display part (14) displays the object received from repository management part (2) through repository I/O part (15). In Step S31, the user pastes the object displayed by the object display part onto the editor using Web page layout part (16), and lays out the Web page of "OX Ski Area Information" in Figure 7. In Step S34, the configuration information of the Web page is stored in repository management part (2) when the layout is completed. Configuration of the Object DB. /7

Figure 14 shows the configuration of object DB (8) in repository management part (2). Object DB (8) shown in Figure 14 is comprised of the fields of object ID, original URL, header anchor tag, footer anchor tag, object type, the object itself, last date and time updated, and update frequency. The weather map object in Figure 7, for example, is stored in object DB (8) as the object ID "A000001" (204), the original URL "http://ab/c.htm" (205), the header anchor tag "<p Weather Map of OX Region </p>" (206), the footer anchor tag "<p><big><big> Lodging Reservation Status" (213), the footer anchor tag "</table>" (214), the object type "table" (215), the object itself of "table data (HRML)" (216), the last date and time updated of "1997.11.1.12:00" (217), and the update frequency of "every day at 0:00 and 12:00" (218). Thus, "header anchor tags" (206) and (213) in Figure 14 reflect the revisions, including text strings, made by the user using object extraction editor (4). It will also be noted that the user can specify the update frequency of each object pasted by the Web page generation editor, and this is recorded in update frequency (219) and (218) in object DB (8) in Figure 14.

#### Configuration of Web Page Configuration information DB

Figure 15 shows the configuration of Web page configuration information DB (7) in repository management part (2). Web page configuration information DB (7) with attribute (15) is comprised of the merging Web page URL name, the merging Web page layout information (HTJL format), and an update flag. The table object in Figure 7, for example, is stored in Web page configuration information DB (7) as the merging Web page URL name "http://MyServe/ski.htm" (220) and the merging Web page layout information (HTML format) of "layout" (221), with the update flag

"ON". Web page generation editor (5) determines the update information (whether or not updated when viewed) for objects in the generated Web page. In this example, update flag (222) in Figure 15 is ON because the user "updated when viewing."

Figure 19 is a diagram showing the layout information of the informational Web page. Figure 19 shows the "layout information" stored in merging Web page layout information (HTML format) field (221) in Figure 15. (238) and (239) in Figure 19 correspond to "<C\_P\_ObjectID=A000001>" and "<C\_P\_ObjectID=A000001>" described in object ID fields (201) and (211) in Figure 14.

### Viewing Web Pages

Next, the operation of viewing a Web page will be explained based on the operational flow charts shown in Figures 11, 12, and 13. Figure 11 is a diagram showing a flow chart illustrating the operation when viewing a merging Web page in Working Example 1, and Figure 12 is a diagram showing a flow chart illustrating the operation of Web page execution part (3) in Figure 11.

In Step S35 in Figure 11, the user inputs the URL for "Ox Ski Information" in Web browser (21). In Step S36, Web server (20) receives the request from Web browser (21), and sends the URL to Web page execution part (3).

### Operation of the Web Page Execution Part

Execution control part (10) searches merging Web page configuration information DB (8) in repository management part (2), and reads the configuration information for the Web page corresponding to the URL name shown in field (220) in Figure 15. As shown in detail in Figure 12, in Step S37, Web page execution part (3) searches merging Web page configuration information DB (8) in repository management part (2), and reads the configuration information for the Web page corresponding to the URL name shown in field (220) in Figure 15. In Step S38, it determines whether or not update flag field (222) in the Web page configuration information is "ON." If update flag field (222) is "ON," processing moves to Step S41. In Step S41, execution control part (10) asks repository management part (2) through repository I/O part (53) to update the object ID ("A00001" ((238) in Figure 19) and "A00002" ((239) in Figure 19) present in the layout information (Figure 19) in the Web page configuration information.

If update flag field (222) is not "ON" in Step S38, execution control part (10) sends the merging Web page configuration

information to Web page relay layout part (9), and asks it to generate a Web page. In Step S40, Web page relay layout part (9) reads the objects on the merging Web page configuration information sent from repository management part (2) through repository I/O part (52), and generates a Web page. The Web page generated in Step S40 is displayed by Web browser (21) through Web server (20) in Step S41. /8

#### Operation of the Repository Management Part

Figure 13 is a diagram showing a flow chart illustrating the operation of repository management part (2) in Figure 11. When repository management part (2) is asked to update object ID ("A00001" ((238) in Figure 19) and "A00002" ((239) in Figure 19) from execution control part (1) in Step S41 in Figure 12, in Step S43, object check engine (6) searches object DB (8) using the key "object ID," and obtains the original URL in the attribute information for the object and the rules (header and footer anchor tags) for extracting the object. In Step S44, object comparison part (17) connects the generated Web page and the Web page indicated by the original URL. In Step S45, it is determined whether or not the date the original Web page was generated is later than the last date and time updated in object DB (8) ((210) in Figure 14). If the date the original Web page was generated is later than the last date and time updated in object DB (8), in Step S46, HTML analysis engine (11) reads the object from the original URL based on the rules (header and footer anchor tags). In Step S47, the read object is rerecorded in object DB (8) from repository I/O part (54). In Step S48, object check engine (6) notifies Web page execution part (3) that checking is completed. If the date the original Web page was generated is later than the last date and time updated in object DB (8) in Step S45, processing also moves to Step S48 and object check engine (6) notifies Web page execution part (3) that checking is completed.

Returning to Figure 12, when Web page execution part (3) has been notified by repository management part (2) that checking is completed, in Step S39, execution control part (10) sends the Web page layout information (Figure 19) read from the repository management part earlier to HTML generation engine (18), and asks it to generate a Web page. In Step S40, HTML generation engine (18) generates the Web page in Figure 7 (HTML source in Figure 20) by entering the object ID part ((238) and (239) in Figure 19) of the Web page layout information in the update object read from the object DB. the Web page generated in Step S40 is displayed by Web browser (21) through Web server (20) in Step S42 of Figure 11.

## [Effects of the Invention]

According to the first and twelfth inventions, this invention is constituted by a Web page generation part extracting objects from several Web pages and generating a new Web page in a chosen layout, a repository management part storing the configuration information of the generated Web page and updating affixed objects, and a Web page execution part generating an actual Web page from the configuration information and objects stored in the above-mentioned repository management part. As a result, required objects can be cut from several Web pages, and a new Web page can be generated following the layout preferred by the user. Furthermore, the objects placed on the generated Web page are automatically updated on the new Web page to the latest objects when the objects on the original Web pages are updated. Consequently, this invention has the effect that information scattered on the Internet can be viewed in a chosen layout and kept updated to the latest information.

According to the second and thirteenth invention, the Web page generation part of this invention is constituted by an object extraction editor extracting objects from several existing Web pages, and a Web page generation editor editing extracted objects. As a result, a new Web page can be easily generated from existing Web pages.

According to the third and fourteenth inventions, the object extraction editor of this invention is constituted by extracting objects from existing Web pages by an HTML analysis part dividing an inputted Web page into objects, and an object revision part revising how objects divided by the above-mentioned HTML analysis part have been divided. As a result, objects can be coarsely extracted from existing Web pages, then how objects have been extracted can be finely revised by user commands. Therefore, objects can be extracted efficiently from existing Web pages.

According to the fourth and fifteenth inventions, the HTML analysis part of this invention is constituted by an analysis rules DB storing rules for extracting objects from Web pages, and an HTML analysis engine dividing an inputted Web page into given objects according to the rules stored in the above-mentioned analysis rules DB. As a result, the operation of revising objects extracted from existing Web pages can be performed by a tactile operation such as a mouse, making the object extraction revision operation by the user more efficient. In addition, attribute information can be appended to objects.

According to the fifth and sixteenth inventions, the object revision part of this invention is constituted by an object package part displaying Web pages divided into objects by the above-mentioned HTML analysis part to the user, revising how objects have been divided, and appending attribute information to objects. As a result, extraction rules can be added to the analysis rules DB when extracting new types of objects unknown before now.

According to the sixth and seventeenth inventions, the Web page generation editor of this invention is constituted by an object display part recording and reading object information and the configuration information of the generated merging Web page to and from the repository management part through a repository I/O part and offering extracted objects to the user as parts for generating a Web page by displaying extracted objects graphically, and a Web page layout part generating a merging Web page by laying out objects displayed by the above-mentioned object display part. As a result, Web pages can be generated with improved ease of operation. /9

According to the seventh eighteenth inventions, this invention is constituted by the repository management part managing objects on an extracted Web page and object attribute information in an object DB, managing the layout information of a generated Web page in a merging Web page configuration information DB, and processing such as updating objects by an object check engine. As a result, the layout of a Web page and the objects on it can be managed independently. Consequently, the user can use one object in several merging Web pages, and need not pay special attention to matters such as matching objects. In addition, the page generation method (such as generating after objects have been updated) can be set by page units using update flag information. Furthermore, being able to manage objects on scattered Web pages in a centralized manner facilitates checking updates and the like, and improves the reusability of used Web pages.

According to the eighth and nineteenth inventions, the object check engine in this invention is constituted by an HTML analysis engine recording and reading object information to and from the repository management part through a repository I/O part and analyzing, extracting, and sending objects to the object comparison part according to the extraction rules in the read object information, and an object comparison part comparing objects recorded in the repository management part sent from the repository I/O part to original objects sent from the HTML analysis engine and requesting the repository I/O part to rerecord an object in the repository management part if the object has been updated. As a result, objects on Web pages can be automatically checked for updates, merged, and rerecorded. Therefore, the user can always view the latest

information without being aware of it. In addition, by performing updates in the background, a merging Web page can be generated without checking objects when viewed, thereby improving response when viewing.

According to the ninth and twentieth inventions, the Web page execution part of this invention is constituted by an execution manager for reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and controlling the repository management part and the Web page layout part based on the read object information and merging Web page configuration information, the above-mentioned HTML generation engine part recording and reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and generating an HTML file based on the read object information and Web page layout information. As a result, Web pages can be generated dynamically from the repository of merging Web pages.

According to the tenth invention, the Web page execution part in this invention is comprised of an HTML generation engine part for recording and reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and generating an HTML file based on the read object information and Web page layout information. As a result, Web pages can be generated dynamically from the repository of merging Web pages. Therefore, the user need not be concerned with matters such as updating Web pages.

According to the eleventh invention, the execution control part in this invention is comprised of an execution manager for reading object information and the configuration information of a merging Web page to and from the repository management part through a repository I/O part and controlling the repository management part and the Web page layout part based on the read object information and merging Web page configuration information, and controls generation of Web pages. As a result, the sequence for generating a merging Web page can be modified according to the properties of the merging Web page (such as updated periodically or updated when viewed), and merging Web pages can be viewed efficiently. In addition, objects on scattered Web pages can be managed in a centralized manner, thereby improving reusability.

[Brief Explanation of the Figures]

[Figure 1] Diagram showing the configuration of a Web page

generation device in a working example of this invention

[Figure 2] Diagram showing the configuration of the Web page generation part in a working example of this invention

[Figure 3] Diagram showing the configuration of the Web page execution part in a working example of this invention

[Figure 4] Diagram showing the configuration of the repository management part in a working example of this invention

[Figure 5] Diagram showing an original Web page with a weather map in a working example of this invention

[Figure 6] Diagram showing an original Web page with lodging information in a working example of this invention

[Figure 7] Diagram showing a merging Web page (informational Web page) generated in a working example of this invention

[Figure 8] Diagram showing a flow chart illustrating the operation when generating a merging Web page in a working example of this invention

[Figure 9] Diagram showing a flow chart illustration the operation of the object extraction editor in a working example of this invention

[Figure 10] Diagram showing a flow chart illustrating the operation of Web page generation editor in a working example of this invention /10

[Figure 11] Diagram showing a flow chart illustrating the operation when viewing a merging Web page in a working example of this invention

[Figure 12] Diagram showing a flow chart illustrating the operation of the Web page execution part in a working example of this invention

[Figure 13] Diagram showing a flow chart illustrating the operation of the repository management part in a working example of this invention

[Figure 14] Diagram showing the table contents of the object DB in a working example of this invention



[Figure 15] Diagram showing the table contents of the merging Web page configuration information DB in a working example of this invention

[Figure 16] Diagram showing the table contents of the analysis rules DB in a working example of this invention

[Figure 17] Diagram showing the source of an original Web page with a weather map in a working example of this invention

[Figure 18] Diagram showing the source of an original Web page with lodging information in a working example of this invention.

[Figure 19] Diagram showing the layout information of an informational Web page in a working example of this invention

[Figure 20] Diagram showing the sources for an informational Web page in a working example of this invention

[Figure 21] Diagram showing the method for viewing Web pages using a link assembly by prior art

[Explanation of Reference Numerals]

1 ... Web page generation part, 2 ... repository management part, 3 ... Web page execution part, 4 ... object extraction editor, 5 ... Web page generation editor, 6 ... object check engine, 7 ... Web page configuration information DB, 8 ... object DB, 9 ... Web page relayout part, 10 ... execution control part, 11 ... HTML analysis engine, 12 ... analysis rules DB, 13 ... object package part, 14 ... object display part, 15 ... repository I/O part, 16 ... Web page layout part, 17 ... object comparison part, 18 ... HTML generation engine, 19 ... execution manager, 20 ... Web server, 21 ... Web browser, 50 ... HTML analysis part, 51 ... object revision part, 52 ... repository I/O part, 53 ... repository I/O part, 54 ... repository I/O part

Figure 1

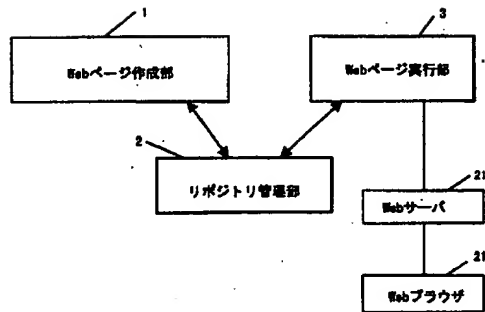


Figure 2

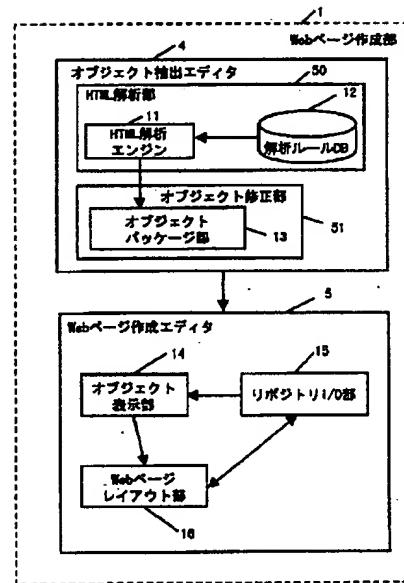


Figure 8

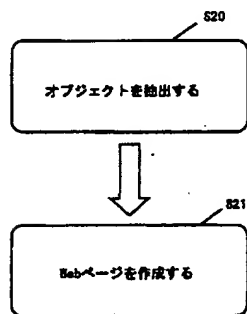


Figure 1

Key: 1: Web page generation part; 2: repository management part; 3: Web page execution part; "21"[should be 20]: Web server; 21: Web browser.

Figure 2. Configuration of Web page Execution Part

Key: 1: Web page generation part, 4: object extraction editor; 5: Web page generation editor; 11: HTML analysis engine; 12: analysis rules DB; 13: object package part; 14: object display part; 15: repository I/O part; 16: Web page layout part; 50: HTML analysis part; 51: object revision part.

Figure 8

Key: S20: extract objects; S21: generate Web page.

Figure 3

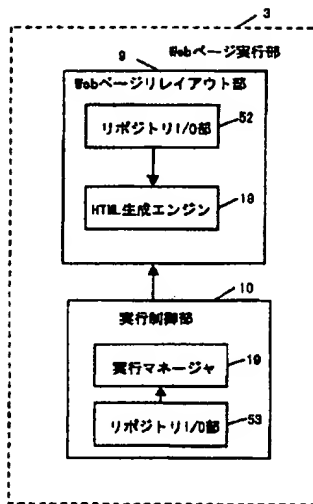


Figure 5

Figure 4

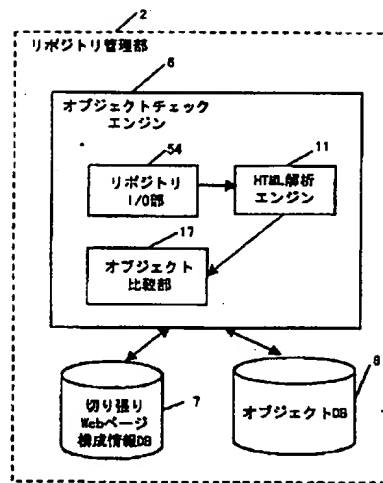


Figure 6

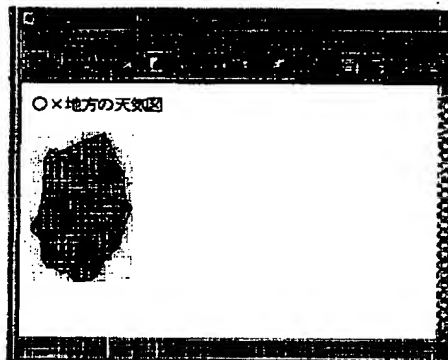


Figure 15

〇×スキー場お宿情報 (a)

ごあいさつ (b)

みくそ. (c)

宿の予約状況 (d)

(e) 宿名/日時	(f) 12月13日	(g) 12月14日
〇×ロッジ	○	○
〇×旅館	△	○
〇×ペンション	×	△

(k) スキーー口メモ

(l) スキーは2日で済みます。♪

Figure 21

切り張りWeb ページURL名	Webページレイアウト 情報 (HTML形式)	更新フラグ
http://MyServe/s ki.htm	レイアウト情報	ON

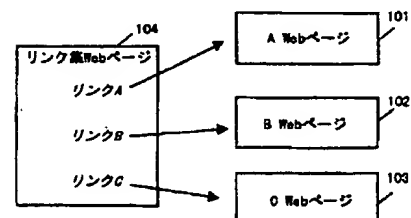


Figure 3

Key: 3: Web page execution part, 9: Web page relayout part, 10: execution control part, 18: HTML generation engine, 19: execution manager, 52: repository I/O part, 53: repository I/O part

Figure 4

Key: 2: repository management part, 5: Web page generation editor, 7: Web page configuration information DB, 8: object DB, 11: HTML analysis engine, 17: object comparison part, 54: repository I/O part

Figure 5

Key: Weather Map of OX Region

Figure 6

Key: a: Lodging Information in the OX Region, b: Greetings, c: Welcome, d: Lodging Reservation Status, e: Name of Lodge / Date, f: December 13, g: December 14, h: OX Lodge, i: OX Inn, j: OX Pension, k: Brief Ski Memo, l: "Ski" means sliding on your knees.

Figure 15. Contents of Merging Web Page Configuration Information

Key: Above 220: URL Name of Merging Web Page, Above 221: Web Page Layout Information (HTML format), Above 222: Update Flag, 221: *layout information*

Figure 21

Key: 101: Web page A, 102: Web page B, 103: Web page C, 104: Link Assembly Web Page: link A, link B, link C

Figure 7

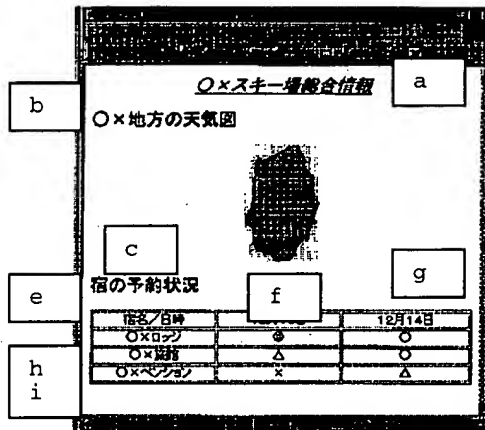


Figure 9

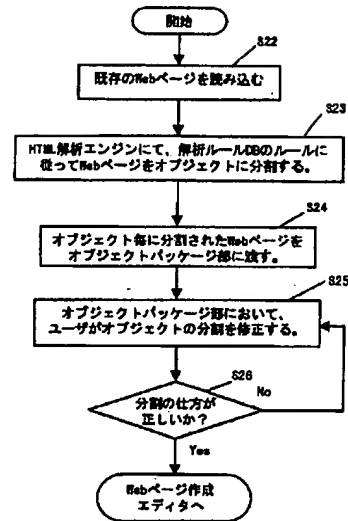


Figure 11

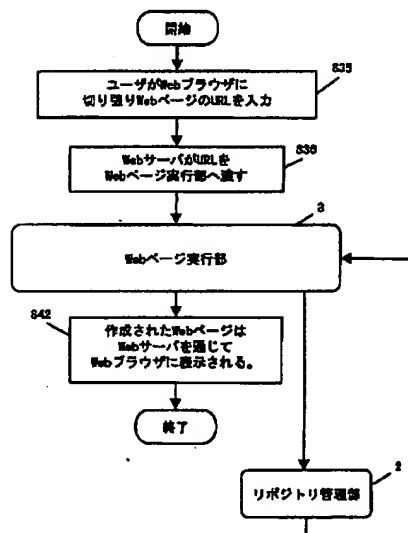


Figure 16

オブジェクトタイプ	先頭アンカータグ	終端アンカータグ
図	<p><img src=	></p>
表	<table	<table>

Figure 7

Key: a: OX Ski Area Information, b: Weather Map of OX Region, c: Lodging Reservation Status, d: Name of Lodge / Date, e: December 13, f: December 14, g: OX Lodge, h: OX Inn, i: OX Pension

Figure 9. Flow Chart Showing Operation of Object Extraction Editor

Key: Top: start, S22: read existing Web page, S23: divide Web page into objects by HTML analysis engine following rules in analysis rules DB, S24: send Web page divided into objects to object package part, S25: user revises division of objects in object package part, S26: is how objects are divided correct?, Bottom: to Web page generation editor

Figure 11

Key: 2: repository management part, 3: Web page execution part, Top: start, S35: user inputs URL of merging Web page in Web browser, S36: Web server sends URL to Web page execution part, S42: display generated Web page by Web browser through Web server, Bottom: end

Figure 16. Table Contents of Analysis Rules DB

Key: Above 223: Object Type, Above 224: Header Anchor Tag, Above 225: Footer Anchor Tag, 223: map, 226: table

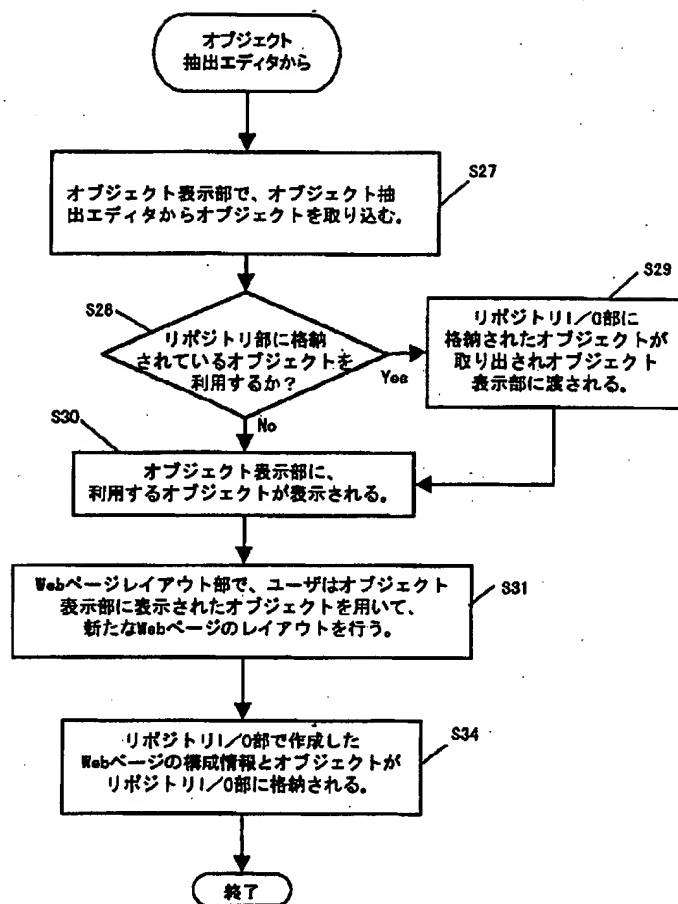


Figure 10. Flow Chart Showing Operation of Web Page Generation Editor  
 Key: Top: from object extraction editor, S27: read objects from object extraction editor by object display part, S28: use objects stored in repository part?, S29: read objects stored in repository I/O part and send to object display part, S30: display objects to be used by object display part, S31: user lays out new Web page by Web page layout part using objects displayed by object display part, S34: store configuration information of Web page generated by repository I/O part and objects in repository I/O part, Bottom: end



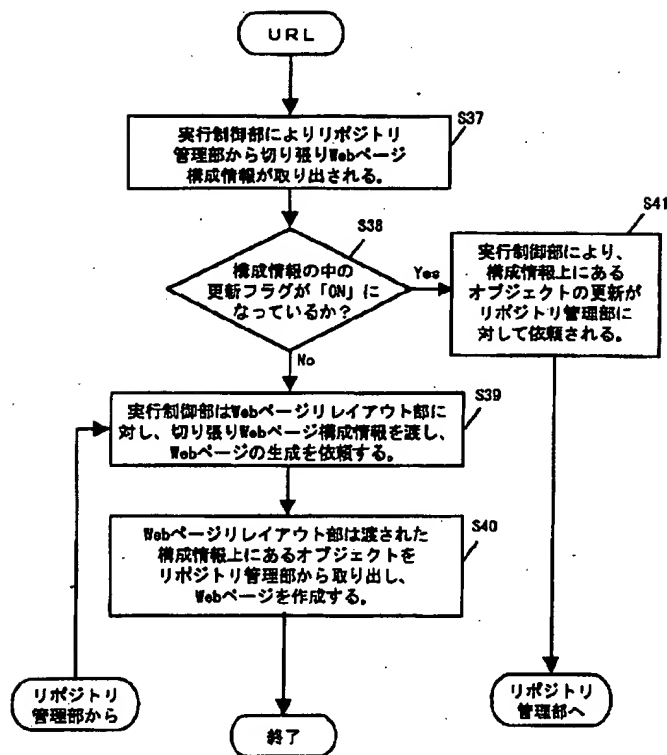


Figure 12. Flow Chart Showing Operation of Web Page Execution Part  
 Key: S37: read configuration information of merging Web page from repository management part by execution control part, S38: is update flag in configuration information "ON"?, S39: execution control part sends configuration information of merging Web page to Web page layout part and asks it to generate Web page, S40: Web page layout part reads objects on the sent configuration information from repository management part and generates Web page, S41: execution control part asks repository management part to update certain objects on configuration information, Lower Left: from repository management part, Lower Right: to repository management part, Bottom: end

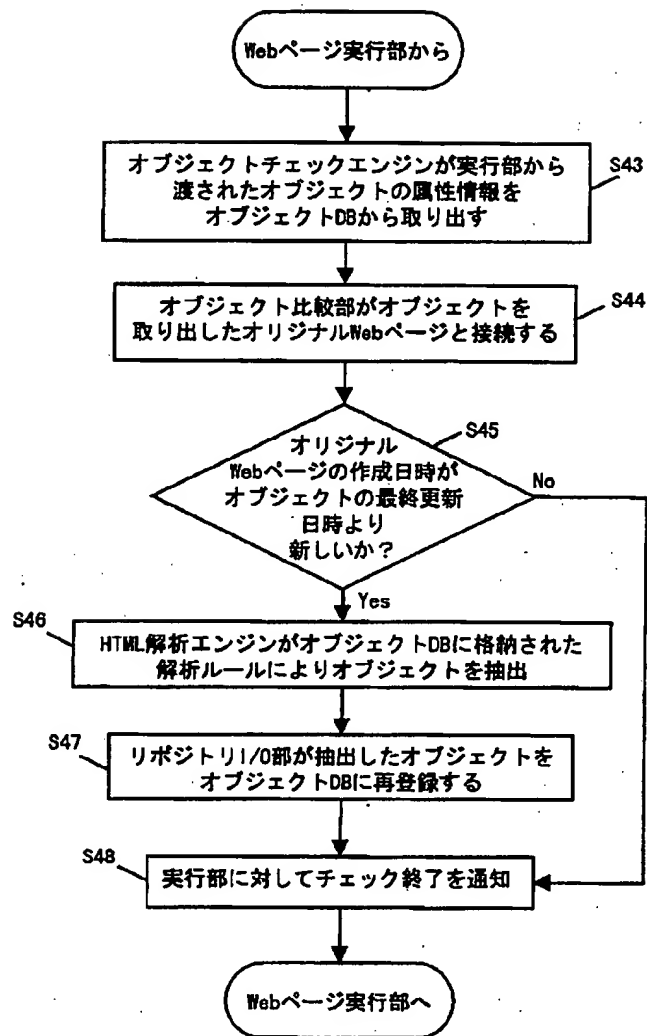


Figure 13. Flow Chart Showing Operation of Repository Management Part  
 Key: Top: from Web page execution part, S43: object check engine reads attribute information of objects sent by execution part from object DB, S44: object comparison part connects objects to original read Web page, S45: date and time that original Web page was generated later than date and time of last update of objects?, S46: HTML analysis engine extracts objects by analysis rules stored in object DB, S47: reregister extracted objects in repository I/O part in object DB, S48: notify execution part that check completed, Bottom: to Web page execution part

オブジェクトID	オリジナルURL	先頭アンカータグ	終端アンカータグ	オブジェクトタイプ	オブジェクト本体	最終更新日時	更新頻度
A0000001	http://ab/o.htm	<p>O×地方の 天気図</p>	<p>", 208: map, 209: weather map data (JPG), 213: <p><big><big> Lodging Reservation Status, 215: table, 216: table data (HTML), 218: every day 0:00, 12:00, 219: every day 13:00				

Figure 17

```

<html>

<head>
<meta http-equiv="Content-Type" content="text/html; charset=ujis">
<title>ホーム ページ</title>
</head>

<body>
<p>〇×地方の天気図</p>
<p></p>

<p></p>
</body>
</html>

```

Figure 18

```

<<中略>> (a)
<p align="center"></p>
<p align="center">〇×スキー場お国情報</p>
<p align="center">〇×地方の天気図</p>
<p align="center"></p>
<table border="1" width="100%">
<tr>
<td width="33%" align="center">〇×スキー場お国情報 (f)
<td width="33%" align="center">1 2月1 3 日 (g)
<td width="34%" align="center">1 2月1 4 日 (h)
</tr>
<tr>
<td width="33%" align="center">〇×スキー場お国情報 (i)
<td width="33%" align="center">〇×スキー場お国情報 (j)
<td width="34%" align="center">〇×スキー場お国情報 (k)
</tr>
<tr>
<td width="33%" align="center">〇×スキー場お国情報 (l)
<td width="33%" align="center">〇×スキー場お国情報 (m)
<td width="34%" align="center">〇×スキー場お国情報 (n)
</tr>
</table>
</p>
</body>
</html>

```

Figure 20

```

<<中略>> (a)
<p align="center"></p>
<p align="center">〇×スキー場お国情報</p>
<p align="center">〇×地方の天気図</p>
<p align="center"></p>
<table border="1" width="100%">
<tr>
<td width="33%" align="center">〇×スキー場お国情報 (f)
<td width="33%" align="center">1 2月1 3 日 (g)
<td width="34%" align="center">1 2月1 4 日 (h)
</tr>
<tr>
<td width="33%" align="center">〇×スキー場お国情報 (i)
<td width="33%" align="center">〇×スキー場お国情報 (j)
<td width="34%" align="center">〇×スキー場お国情報 (k)
</tr>
<tr>
<td width="33%" align="center">〇×スキー場お国情報 (l)
<td width="33%" align="center">〇×スキー場お国情報 (m)
<td width="34%" align="center">〇×スキー場お国情報 (n)
</tr>
</table>
</p>
</body>
</html>

```

Figure 17. HTML Source of Figure 5

Key: <title>: Home Page, 230: <p> Weather Map of OX Region </p>,  
231: ... (2216 bytes) ...

Figure 18. HTML Source of Figure 6

Key: a: <<omitted>>, b: Lodging Information in the OX Region, c:  
Greetings, d: Welcome, e: Lodging Reservation Status, f: Name of  
Lodge / Date, g: December 13, h: December 14, i: OX Lodge, j: OX  
Inn, k: OX Pension, l: Brief Ski Memo, m: "Ski" means sliding on  
your knees

Figure 20. HTML Source of Figure 7

Key: a: <<omitted>>, b: OX Ski Area Information, c: Weather Map of OX  
Region, d: Lodging Reservation Status, e: Name of Lodge / Date,  
f: December 13, g: December 14, h: OX Lodge, i: OX Inn, j: OX  
Pension

```

<html>
<head>
 <<中略>> (a)
</head>
<body>
 <p align="center"></p>
 <div align="center"><center>

 </center></div>
</body>
</html>

```

Figure 19 shows the HTML source code for Figure 7. The code is enclosed in a box. It starts with <html>, followed by <head> and </head>. Inside <head>, there is a comment <<中略>> (a). The <body> section contains a paragraph <p align="center"> with an <img> tag. The <img> tag has an alt attribute with the value "Ox Ski Area Information" (labeled (b)), a src attribute with the value "C\_P\_ObjectID=A000001", a width attribute with the value "99", and a height attribute with the value "158". Below the paragraph, there is a <div align="center"> block containing a <center> block. Inside the <center> block, there is another <img> tag with an alt attribute with the value "C\_P\_ObjectID=A000002". The code ends with </div>, </body>, and </html>.

Figure 19. HTML Source of Figure 7

Key: a: <<omitted>>, b: Ox Ski Area Information